DRAGON USER



The independent Dragon magazine

75p US$3.25 June 1984

Play Grand Prix in fast and furious machine code

Talking in tongues

Dragon books put through their paces

Games review brings in the winners

WIN Canon drive from Alpha Disc

NEW RELEASES FOR THE DRAGON 32/64 FROM SALAMANDER SOFTWARE

TURTLE GRAPHICS

This program has been designed to provide a method of drawing pictures with your computer which is both fun and educational, and yet simple enough to be enjoyed by the young or inexperienced computer user and the enthusiast alike.

Pictures are drawn by telling the "turtle" to move forward and turn so that it traces a picture on your TV screen or 4 colour printer/plotter.

Powerful "repeat" and "word library" facilities allow you to build up new commands as you progress, and you can save your "word library" on cassette tape or floppy disc.

The end result of using a system such as this is an increased knowledge of geometry, and a basic understanding of simple programming, making the program an ideal educational tool for all ages.

- Snappy 32 page manual included.
- Runs from cassette or disc (using either Dragon Data DOS or Premier Microsystems DOS).
- Simple command structure.
- Repeat command sequences – can be nested to 32 levels.
- "Word" definition – can be nested to 30 levels.
- Powerful "word library" handling.
- Save and load library using cassette or disc.
- Dynamic editing of command strings and word library.
- Scaling and stretching commands.
- Will draw pictures on 4 colour printer/plotter (TRS-80 GCR115, MCP-40, or compatible plotter).

Price £9.95



DATA RETRIEVAL SYSTEM

DRS is a powerful database package for the Dragon computer, supporting files stored either on cassette or disc (using the Dragon Data DOS).

DRS has been designed for ease of use and flexibility. You can design your own database, specifying number and type of fields. Powerful full screen editor allows easy design of your own screens.

DRS contains full facilities for adding, maintaining and deleting records. Fields may be added to or deleted from existing files even after records have been added. Search facilities allow for selection of records by any portion or combination of fields.

DRS also has an extremely flexible report generator, allowing for anything from printing of labels to paged reports.

- *22 000 characters of data per file.*
- *154 000 characters of data per disc file.*
- *Each record may contain a maximum of 480 characters, 26 alphanumeric fields and 10 numeric fields.*
- *Inclusive, exclusive and ranged searches on alphanumeric fields.*
- *Comparative searches on numeric fields.*
- *Add, delete and change records.*
- *User definable screen displays.*
- *User definable report generator.*
- *Record formats may be changed at any time.*
- *Disc files require Dragon Data DOS.*

Price £14.95



Tape duplication by DATACLONE

Please add 50p P.&P. to all orders. Send A5 SAE for full catalogue.
Cheques or postal orders payable to:

Salamander SOFTWARE

17 Norfolk Road, Brighton, BN1 3AA. Telephone: Brighton (0273) 771942.

Look out for these other new releases from Salamander Software.

DRAGON: **Wings of War £7.95** (WW II Adventure). **The Cricklewood Incident £7.95** (Weird & Wonderful Adventure).

BBC: **Eagle £7.95** (Original Arcade Action). **Turbo £9.95** (BASIC Compiler)



Telephone number
(All departments)
01-437 4343
Editor
GRAHAM CUNNINGHAM
Assistant Editor
GORDON ROSS
Software Editor
GRAHAM TAYLOR
Editorial Secretary
CLEO CHERRY
Advertisement Manager
DAVID LAKE
Advertisement Executive
SIMON LANGSTON
Administration
THERESA LACY
Managing Editor
DUNCAN SCOT
Publishing Director
JENNY IRELAND

Subscriptions
UK £10 for 12 issues
Overseas (surface) £16 for 12 issues

ISSN 0265-0177. Telex: 296275
Dragon User, 12/13 Little Newport Street, London WC2R 3LD.

US address: c/o Business Press International, 205 East 42nd St, New York, NY 10017.

Published by Sunshine Books, Scot Press Ltd. ©Sunshine Books 1984.
Typesetting by Chesham Press, Chesham, Bucks. Printed by Eden Fisher (Southend) Ltd, Southend-on-Sea, Essex.
Distributed by S.M. Distribution, London SW9. 01-274 8611. Telex: 261643

Registered at the Post Office as a newspaper.

How to submit articles

The quality of the material we can publish in *Dragon User* each month will, to a very great extend, depend on the quality of the discoveries that you can make with your Dragon. The Dragon 32 computer was launched on to the market with a powerful version of Basic, but with very poor documentation.

Every one of us who uses a Dragon will be able to discover new tricks and quirks almost every day. To help other Dragon users keep up with the speed of the development each of us must assume that we made the discovery first — that means writing it down and passing it on to others.

Articles which are submitted to *Dragon User* for publication should not be more than 3000 words long. All submissions should be typed. Please leave wide margins and a double space between each line. Programs should, whenever possible, be computer printed on plain white paper and be accompanied by a tape of the program.

We cannot guarantee to return every submitted article or program, so please keep a copy. If you want to have your program returned you must include a stamped, addressed envelope.

Contents



Editorial

WHAT HAVE JIFFY Signs, Space Communications and Dragon Data got in common? Well, one answer is that they were all displaying their respective wares at a recent conference for electrical dealers.

Also at the conference was Dragon Data's managing director Brian Moore, giving a glimpse into the future of information technology — and of Dragon Data itself. Brian's vision of the future is exciting. The various technologies in the home, including microcomputers, are expected to be combined as "multi-function products". So, for example, "computers will increasingly have to contain communications hardware as a basic requirement" and "will increasingly be involved in controlling home services, eg security lighting and heating". And Dragon Data is developing its new products with this trend in mind.

But the path to the future is not lined solely with roses — there will be some casualties along the way. "Undoubtedly, there will be a shake out of manufacturers over the next few years," Brian says, doubtless excluding Dragon Data itself from this particular trend. In his speech to the electrical dealers Brian concentrated on the threat of the Japanese micros running under the same operating system, MSX, written by Microsoft whose Basic is used on the Dragon. As the micros share the same operating system, software for one MSX machine should run on any other — although the first MSX micro to reach the UK reportedly needs modifications to reach compatibility. This strategy "could theoretically dominate major sectors of the home computer market". But, as Brian also said, "it remains to be seen how the UK market will react to the strong marketing presence of the major Japanese producers". It also remains to be seen how the activities of UK firms will affect Dragon Data. Sinclair made a success of following an independent path with the Spectrum, and looks set to do so again with the QL. And fresh entries are being made by companies new to the micro market. For example, the Amstrad micro, including a monitor and built-in cassette recorder, is expected to be sold by Boots, whose stores have sold so many Dragons.

Identifying why people buy a particular micro is as difficult as predicting who will avoid the "shake out". Flatteringly, Brian attributes some influence on purchasing to the power of the press — although he puts it in less flattering terms: "Buying decisions are influenced by fashion and a fair degree of media hype. A large and growing specialist press produces a barrage of advice and product evaluation." Ah well, we'll try to maintain the barrage, although it's little consolation to hear that "it is very fashionable to be writing about home computers". Excuse us while we go away to check our perms and fetch our leather trousers from the cleaners.

Letters

This is the chance to air your views — send your tips, compliments and complaints to Letters Page, *Dragon User*, 12-13 Little Newport Street, London WC2R 3LD.

Unsolved adventure

IN YOUR March issue John Scriven reviewed an arcade plus adventure game — Death Mines of Sirus by Phoenix Software.

Having failed to complete the arcade game to find the loading code for the adventure, John left the game unsolved. If he had succeeded he would have found that the second part of the adventure is not "text only" but is in fact an interesting graphic maze.

Brett Spencer
Ilford
Essex

To the rescue

ONE OF our Dragon 32 titles, Death's Head Hole, simulates the work of a cave rescue team.

A customer of ours, Mr D S Brinkman of 2 Park Street, Denbigh, Clwyd, is a teacher training student, studying the feasibility of computer-assisted learning in the school's outdoor education curriculum. He would be grateful to hear from anyone who has used Death's Head Hole in a school, or as part of a club training scheme.

Harry Whitehouse
Peaksoft
Burton-on-Trent

POKEing around

THE FOLLOWING POKEs can quite effectively be used to define your own cursor:

```
POKE 364, 167: POKE 365, 159:
     POKE 366, 0: POKE 367,
     136
POKE 363, (ASCII code of char-
     acter)
POKE 362, 134 (to activate)
POKE 362, 57 (to de-activate).
```

We all know by now the famous POKEs for disabling the BREAK key for program protection, but the RESET button may be pressed to exit your program and making the machine crash when RESET is pressed is not really satisfactory. So the following program causes the program currently in RAM to be run.

```
10 FOR A=&H200 to &H218
20 READ A$
30 POKE A,VAL("&H"+A$)
40 NEXT
50 DATA 12, 86, 55, 97, 71, 8E,
   02, 00, 9F, 72, 30, 8C, 06,
   32, 63, 4F, 7E, 83, 7D, 20,
   52, 55, 4E, 20, 00
```

Then type EXEC &H200 and every time the RESET button is pressed the program will run.

Jonathan Hale
Swansea

More hints from hi-fi

WITH REGARD to the Hints from hi-fi letter in your March issue, demagnetising heads is a process I have employed for some time now on all my cassette heads. The process is simple, quick and effective, however, a word of warning to newcomers to this practice: Do not allow the demagnetiser to come any nearer than two feet from any cassette whether or not it is in use. This is because the demagnetiser will do more than its job, and will erase any information stored on the tape. I found out the hard way.

Colin Cowan
East Finchley
London

Too long

AS Brian Cadge stated in reply to a reader in the February edition of *Dragon User*, the values of the joystick are updated only when JOYSTK(0) is used.

The subroutine called when JOYSTK(0) is used resides at $8012. Therefore, rather than using:

```
10 A = JOYSTK(0):? JOYSTK(1):
   GOTO 10
```

which takes twice as long as necessary, try using

```
 0 EXEC:&H8012
10 EXEC:? JOYSTK(1): GOTO10
20 &H8012 = 32786
```

Secondly, if you need a complete memory map then I suggest you contact Hilton Computer Services. I bought one for £5 and I found no less than 611 useful locations.

David Donnelly
Gateshead
Tyne and Wear

Simple stop

A ROUTINE to stop the LIST and LLIST function was given by Brian Cadge in the March edition of *Dragon User*.

This routine was some 14 lines long. A simpler method is to include this one line: POKE 383,158: POKE 157,125: POKE 158,0.

This line can be hidden in the progam with a GOSUB . . . at the beginning of the program. Further, if the program is auto-run, this routine is operated straight away.

Peter Leach
Antrim
N Ireland

Black on orange

I HAVE perfected a short machine code routine which gives the Dragon 32 a black on orange display. The Basic program to load it is as below:

```
10 FOR A = 32760 TO 32765
20 READ B
30 POKE A,B
40 NEXT
50 POKE 363, 127
60 POKE 364, 248
70 POKE 362, 126
80 DATA 134, 12, 183, 255, 34,
   57
```

To disable the effect, type POKE 362, 57 and to re-enable it, type POKE 362, 126.

M A Poyser
Rochester
Kent

Switching on

I REFER to the many queries that your magazine has received regarding the use of tape recorders with the Dragon. When I first tried to use my tape recorder I found a problem which though easy to cure, is common to many cheap recorders.

The problem is with the remote control switch. Most small tape recorders operate so that the switch controls both the motor and the amplifier. When the power is supplied, the amplifier takes a little time to become active. If the computer starts sending information during this period, it will be lost or distorted. The solution is to alter the wiring so that the switch only controls the motor, and the amplifier is left permanently on.

Chris Jolly
Grove Park
London

Mis-hash of Mash

MASH has smashed was the apt comment from one of the many readers who contacted us complaining about the errors in the program listing Mash (*Dragon User*, April).

Unfortunately we have still not been able to remedy this problem program and must apologise for all the frustrating hours many readers have spent attempting to get it to work.

However, we are trying to get the program to work and if we're successful we'll send the corrected listing to readers who have contacted us.

If possible, we'll also publish the new listing. Look out for further announcements on this page.

Once again, our apologies for this program which was only published due to an administrative oversight.

We play readers' programs before accepting them, and then we LIST them — but unfortunately the system broke down for Mash.

Software Top 10

1	(–)	**Dragon Chess**	Oasis Software
2	(6)	**Hungry Horace**	Melbourne House
3	(–)	**Chuckie Egg**	A&F Software
4	(2)	**Ugh**	Softek
5	(–)	**Pedro**	Imagine
6	(3)	**Eightball**	Microdeal
7	(–)	**Kriegspiel**	Beyond
8	(4)	**Cuthbert in the Jungle**	Microdeal
9	(–)	**Up Periscope**	Beyond
0	(7)	**Frogger**	Microdeal
	(–)	**Skramble**	Microdeal

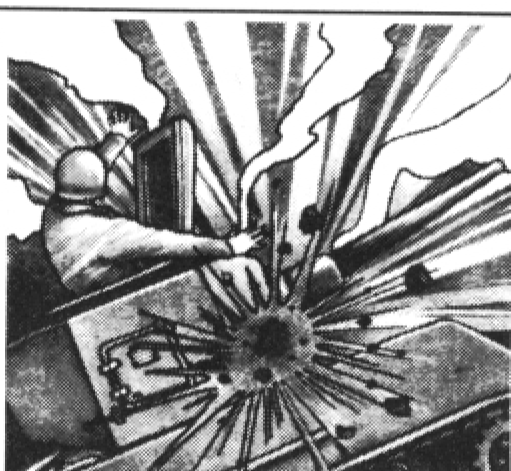
Chart compiled by Websters Software


Games that come from...
BEYOND
CHALLENGING SOFTWARE

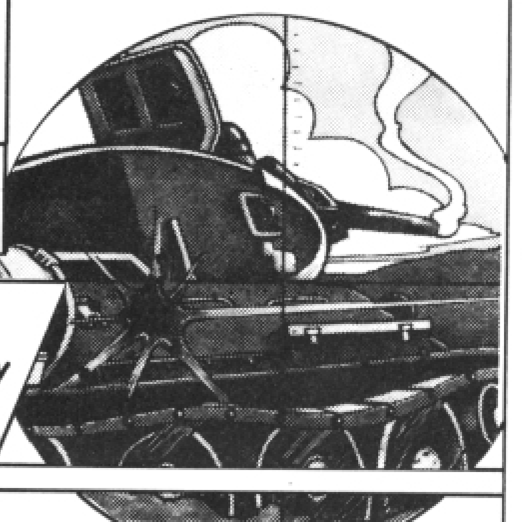

UP PERISCOPE
PROTECT the convoy using SONAR + Depth charges to seek out and destroy the enemy below!


Try and pick off your Enemy from below the waves!


Kriegspiel
Here comes the first flakes of snow and out of it – their Tanks!


...How much longer can we hold this town...?




KRIEGSPIEL:
A thrilling game of strategy to be played against the Dragon or any other devious opponent.



OASIS SOFTWARE
present . . . TWO NEW RELEASES
SPRINT BASIC COMPILER
A rapid step forward in Home Programming.
The newly released 1.1 upgraded compiler written by Dr. David Gray for Dragon 32 and 64 owners.
By converting your own BASIC programs into intermediate code, Sprint accelerates them by up to 5 or even 10 times their normal running speed. Supporting virtually all the integer BASIC and cassette and printer commands, Sprint programs are not only compiled from tape but also run independently of the compiler.
SPRINT
Basic Compiler For the DRAGON 32
OASIS SOFTWARE
DRAGO
PETI PASC
an integer subset of the structured programming language for the DRAGON 32 MICROCOMPUTER
PETITE PASCAL
Not just a language – an education in structured programming.
This extended integer subset of the structured programming language is the ideal introduction, not only to a language widely used in schools and universities but also to programming itself.
As an appreciation of PASCAL is generally accepted as the best way to increase your understanding of structured languages, this is a must for anyone who takes computing seriously.
OASIS SOFTWARE
Oasis Software, 9a Alexandra Parade, Weston-super-Mare, Tel: 0934 419921
Please send me
SPRINT BASIC COMPILER £14.95
PETITE PASCAL £14.95
I enclose cheque/PO for £
Name
Address
OASIS SOFTWARE

DRAGON 32 & 64

DRAGON CHESS

- Six levels of play.
- All legal chess moves including en-passant, castelling and pawn to queen promotion.
- List of previous moves stored which can be displayed or printed.
- Loading and saving of games positions from tape.
- High resolution graphics which can be flipped round to make black or white play from either end.
- Simultaneous text and graphics.
- Sides can be exchanged at any stage.
- Best move hint.
- Set up from any position.
- Opening move library.
- Moves may be taken back and play resumed from any point.
- Change level of play at any point in the game.
- Will adjudicate games between humans.
- Very high standard of play.
- Professional packaging and life-time guarantee.

"...I thought this was an excellent version of the game. Well worth buying." (Home Computer Weekly)

"Dragonchess is without doubt the best value chess on the market today." (Microdeal)



MIND GAMES COMPENDIUM.

All five games for just £19.95

BACKGAMMON

The well known game of backgammon complete with full instructions and computer demonstration for beginners.

REVERSI

At last! Reversi running on your Dragon 32 or 64. 2 levels of play, full instructions and computer demonstration for beginners.

INVADER CUBE

As well as being one of the best games of skill written for the Dragon 32 and 64 it also has some of the best pure machine code graphics we have ever seen.

DOMINOES

Two games in one with full instructions. Hours of frustrating fun this program has 6 levels of play and on the higher levels adapts its play to its assessment of your ability.

Oasis Products are available from Boots and all good software suppliers. Including: Websters, Games & Computers, Microdeal, The Dragon Dungeon, PCS Distribution, Pilonmist, Goal Computers (France), Falck Computers (Scandinavia) and Joylab (Germany).

If your local dealer does not stock these products then let us know his address and we will contact him.



Access orders taken by phone 24 hours a day.



0934 419921

Every product carries a lifetime guarantee. All prices include VAT and p & p.

OASIS SOFTWARE

Oasis Software, 9a Alexandra Parade, Weston-super-Mare, Tel: 0934 419921

Please send me

		Dragon 32	Dragon 64
DRAGON CHESS	£9.95	☐	☐
BACKGAMMON	£5.95	☐	☐
REVERSI	£5.95	☐	☐
DOMINOES	£5.95	☐	☐
INVADER CUBE	£5.95	☐	☐
MIND GAMES COMPENDIUM	£19.95	☐	☐

I enclose cheque/PO for £ ______

NAME ______

ADDRESS ______

DRAGON 32 OWNERS

NEW LOW PRICES

Make your Dragon turn into a real computer with the new Double-Density Delta Disk System.

The Delta Disk System Gives You . . .

- An affordable disk system.
- Powerful Delta disk commands.
- Lets you produce and handle random access files as easily as serial files.
- Random sequential and indexed file handling.
- Simple plug into Dragon there are no HARDWARE MODS needed to run DELTA!
- Easily expandable 180K to 1.4 megabyte ON LINE storage.
- Full range of business utility and games software AVAILABLE NOW!
- The price you see is the price you pay. NO HIDDEN RAM upgrade costs.
- Uses under 2K of user – RAM as DELTA is held in EPROM.
- Enables programmer to easily produce applications Software which automatically starts up and operates without any intervention from the user.

FULL RANGE OF BUSINESS SOFTWARE AVAILABLE. SEND SAE FOR DETAILS.

DELTA CARTRIDGE – contains DELTA disk Operating System, User Manual, demonstration diskette	£120.00
DELTA 1 – DELTA Cartridge, User Manual, a single-sided 40-track (180K) drive plus cable	£284.05
DELTA 2 – as DELTA 1, but with a single-sided 80-track (360K) drive	£332.35
Disk Interface cable (supplied with DELTA 1 or 2)	£9.95
ENCODER 09 assembler/disassembler/editor – integral with DELTA	£24.95
HOME ACCOUNTS full home package for DELTA	£14.95
INFORM – Data Base Management System commissioned especially for DELTA Systems	£39.95

NEW

FLEX FOR THE DRAGON USING DELTA

FLEX is an internationally accepted Disk Operating System. It features dynamic file allocation, random and sequential file handling, executive file capability, auto drive searching, file dating, space compression, user environment control, error messages in English and over 20 commands for disk operation.

A large number of high quality software packages are available to run under FLEX ranging from spreadsheets to word processors, compilers to new languages. FLEX is an elegant, friendly and efficient disk based operating system. FLEX is available now for the 64K DRAGONS using PREMIER's DELTA disk system. It is supplied complete with a 200 page manual. An editor and assembler are both supplied with the package.

FLEX is the registered trade mark of Technical Systems Incorporated. £86.25 inc. VAT and P&P.



DOODLE MASTER

DOODLE-MASTER provides the DRAGON 32 graphics programmer with comprehensive facilities for the definition of graphics shapes.

★ Create high quality shapes/characters which can be used in BASIC or MACHINE CODE programs
★ Fully compatible with ENCODER 09
★ Shapes can be generated and edited many times faster
★ A WRITE facility allows the user to generate a series of program lines containing DATA or FCB statements representing the character/shape, which automatically append themselves to the current program in memory
★ Shape data can be saved to cassette or disk
★ A LINK facility allows shapes to be joined together either vertically or horizontally
★ A PMODE adjust function allows the user to examine the generated shape in another PMODE mode

Price: Cartridge £17.95 DELTA DISK £16.95

TOOLKIT FOR DRAGON 32

PREMIER'S ASTONISHING NEW TOOLKIT FOR THE DRAGON 31 PERFORMS THE FOLLOWING AMAZING FUNCTIONS:

- FULL screen editor allowing copying.
- Eight PROGRAMMABLE KEYS.
- 25 full colour low-resolution graphics screens.
- Full range of ERROR HANDLING commands with several RESUME options.
- OVER SIXTY NEW WORDS fully linked to your DRAGON BASIC.
- VARIABLE GOTO/GOSUB commands.
- Superb TRACE command which uses top right of screen only, thus leaving graphics/text intact.
- SEARCH and REPLACE commands for easier program modification.
- Comes complete with COMPREHENSIVE 40+ page MANUAL.

Available in CARTRIDGE or DELTA enhancement format £29.95 inc.



PREMIER MICROSYSTEMS
208 Croydon Road, Anerley, London SE20 7YX
Telephone 01-659 7131 or 778 1706
Dealer and export orders welcomed.

POSTAGE and PACKING
Tapes 1–2 95p, 3+ £1.25.
Cartridges £2.50
Drives or Disk packages £4.50
ALL PRICES INCLUDE VAT.
Send SAE for full catalogue.
ALLOW 28 DAYS FOR DELIVERY.



On your marks for next micro

THE NEXT micro from Dragon Data is expected to include a built-in modem and 3½-inch disk drives.

The 64K machine, which will run both Microsoft Basic and OS9 programs, comes with one or two 3½-inch Sony drives, as used on more expensive micros such as the Apricot, as well as the modem.

The transportable package, with on-board power supply, is expected to sell for about £700. It will make its public debut at the Consumer Electronics Trade Exhibition at Earls Court in May.

Production samples are expected to be available to the public in August/September, but dealer trials will be conducted first.

At the time of writing little more was known about the new machine. Dragon Data was reluctant to release further information until final details and launch plans had been decided.

Managing director Brian Moore explained that the package is being marketed as part of Dragon Data's belief that micros should be "communications based" — that is future computers will have to contain communications hardware as a basic requirement.

Brian Moore woos the electrical retailers

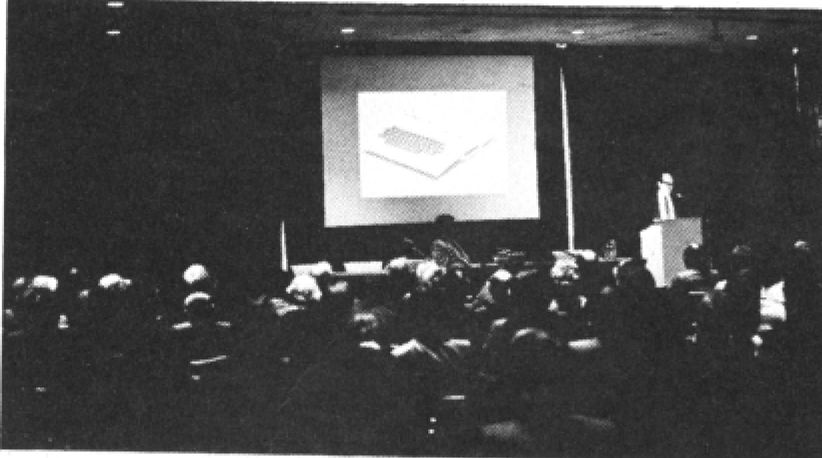
MEMBERS OF the Radio, Electrical and Television Retailers' Association (RETRA) Limited were treated to an informative and light-hearted speech on the home computer market by Brian Moore, managing director of Dragon Data (or is it GEC Dragon?), at their recent annual conference in Torquay.

Speaking about the computer market in general, while displaying Dragon products on the overhead projector, Brian commented that: "The home computer, together with all its support products, represents a retailing opportunity that cannot be ignored. The technology available, together with forecast explosive growth in home communications and information technology, will result in every retailer having to deal in computer-based products in order to survive."

Strong words indeed, and all part and parcel of GEC Dragon's intention to market products through the primary marketers — the independent retailers (see *Dragon User* May 1984).

Brian was not the only speaker to discuss the future implications of advanced technology, several others spoke of the need to sell a complete configuration of equipment — TVs, computers, videos, hi-fis and so on.

Accumulating knowledge of the various systems in order to give customers "expert" advice was also advocated. "I strongly recommend you get



Brian Moore speaking at the RETRA conference

basic keyboard skills," said Brian, imploring retailers to buy a computer, use it for their accounts and then to go around selling it on their personal recommendation.

On the subject of the present home computer business, Brian said it was "totally crazy" consisting of seventeen-year-old millionaires and big businesses failing to make a profit. Businesses producing peripherals, utilities, software and magazines were making money according to Brian, the only people who weren't were the actual computer manufacturers.

This is the key to the computing world. As owners of the Dragon know, the micro on its own is not much use. To be of value, you need a complete package — which is what Dragon Data is now offering. Its display table at the conference consisted of the Dragon 64, a GEC McMichael TV, the OS-9 operating system, Dragon disk drive, joysticks, three cassette-based games, and several pieces of business oriented disk software.

It is the small business user that is targeted as the future growth area. Dragon Data's research has indicated that micros sell for the following five main reasons: games, curiosity/impulse, education, hobbies and business. The latter, it is expected, is the key market with games being the most uncertain one.

Future product specifications should be based on the following criteria: communications based, good "friendly" software, well-packaged, compact and expandable with wider capability.

With this in mind, Brian "unveiled" (on the overhead projector and to the words of a "party political broadcast on behalf of Dragon Data") the new, transportable all-in-one computer package from Dragon Data due out sometime in the near future.

Extra clubs

THE growth in computer clubs continues with the news of several more additions to the Dragon clubnet.

Electronics teacher Tony Walsh has set up the Oldham Computer Users Society at Coldhurst Community Centre, Dunbar Street, Rochdale Road, Oldham. The society has 80 members (23 of whom are Dragon owners) and a Dragon only evening is planned when membership and enthusiasm indicate the necessity.

From Gerard Corcoran comes news of the Ellesmere Port micro group established to promote awareness of microcomputers and their use on Merseyside and to bring together current and potential micro users. The club meets every second Monday and has a regular newsletter. Further details from Gerard on 051-327 3912.

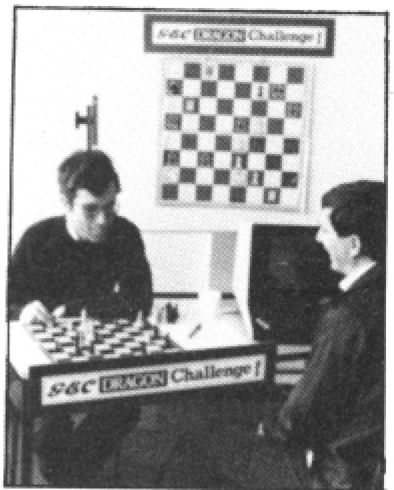
Further afield, Edward Doak brings news of the North Down Microcomputer Users Club in Bangor, Northern Ireland. The club is open to users of different micros though at present the largest group within the club are Dragon users. (There are 61 Dragon 32 users and four members with a Dragon 64 making a total of 65 out of a club membership of 165.)

More information can be obtained from Edward on 0247-50902 and he suggests that any Dragon user thinking of joining should contact him as soon as possible as the club may have to put a ceiling on membership.

Finally from Europe, Maarten Van Wamelen writes about the Dutch section of the Dragon Users club, Oostkamp, Belgium (phone 050-823889) and from Hans Christian Andersen of Andersen Computers, distributor of Dragon Data products in Denmark, comes information of an independent Danish Dragon User club run by Kenneth and Tonnny Christensen, Toften 318, DK-5330, Munkebo, Denmark. (For a complete listing of the Dragon clubnet send a large SAE to *Dragon User*.)

Chess wager

DAVID LEVY laid down the gauntlet — US$5,000 that no computer could defeat him in a chess match. *Omni* science magazine underwrote $4,000 of the wager, GEC Dragon sponsored the tournament and the scene was set for the battle royal between the Cray XMP, one of the most powerful machines in the world and holder of the current world computer chess title, and David Levy, Scottish chess champion, author of the Chess Computer handbook and writer of the Dragon chess program from Dragon Data.



David Levy takes on the Cray.

The challenge matches, held recently at Brunel University, were a victory for mankind. The electronic interloper was soundly defeated and David reckoned it will be another 20 years before a computer will be able to beat a world-class international chess grandmaster. Only time will tell.

Software challenge

FACING Doom, the evil Seiddab or even a putative Dennis Lillee are some of the choices offered by the latest batch of software for the Dragon.

From Paramount Software comes news of Castle of Doom (£6.50), a graphical adventure containing 90 locations and pitting you against the dreaded Count Doom; and Sporting Decathlon, in which you compete in the 10 traditional decathlon events. In the track events you are drawn in lane one racing against two competitors, and in the field events you are the sole competitor.

Hewson Consultants has launched 3D Lunattack, the third game in its space wars epic. Flying a Hoverfighter over the hostile lunar landscape, you have to destroy the Seiddab command base to end the tyranny of the evil Seiddab. The game retails for £7.95 and, as in the other two wars, is played in 3D. Hewson is also intending to flood the French market with its space war series — it has won a contract with the French importers of the Dragon to supply the trilogy to over 300 outlets throughout the country.

Peaksoft presents you with the opportunity to test the wicket in its latest release for the Dragon — Tim Love's Cricket. The game features complete joystick control over batting, bowling and fielding, with the bowling and batting action shown in full-bodied animated graphics. There is a choice of skill levels — Village Green, County and Test matches, a save option to build a library of teams, a bowling practice option, updated bowling averages and scoreboard. Cricket costs £8.95.

Other releases, being given the final finishing touches at the time of going to press, include Polaris Rescue from ACE, distributed by Market Video, and Operation Sasras a "prequel" to Pettigrew's Diary by Shards Software.

Polaris Rescue (£6.90) concerns the rescue attempts of a British team thwarted by the Russians while trying to raise a downed submarine containing a nuclear reactor set to explode at a given time; while Operation Sasras (£7.95) describes the events leading up to the farm house scene in Pettigrew's Diary. It is a three-part adventure using original graphics and making extensive use of another of Shards programs, Shaper, to show the sounds you can obtain from the Dragon.

Disk drive

ALPHA Disc has enhanced the Canon 40/80 track disk drive making it "even easier for unsophisticated users to work, and also extra safe in any teaching environment".

The basic MDD 221 has an LED which shows a green light when the drive is ready for use, and a red light when the head is loaded/drive selected. To this the company has added a built-in 40 or 80 track LED indicator and secondary switching power supply.

The enhancement develops the two signal structure into a four signal structure; a dimmed green light signalling power on, a brilliant green light denoting 40 track select; and likewise a dimmed red light shows that the power is on and a brilliant red light denotes 80 track mode selected.

The secondary switching power supply it is claimed both avoids the problem of heat generation through the standard linear power supply and the expense of screening usually associated with switching systems.

The disk drive with enhancements, the Canon MDD 221AD, costs £224 when operating off the computer's power and £264 for the model with secondary switching power. It is available direct from Alpha Disc at Unit 2, Crabtree Road, Thorpe Industrial Estate, Egham, Surrey.

Grafpad

BRITISH MICRO's Grafpad graphic tablet, originally available for the BBC micro, is now compatible with the Dragon.

The Grafpad (price £125) allows you to draw designs or trace the outlines of maps and pictures. These can then be saved to disk or cassette and copied through a printer.

Based on the ULA chip, the Grafpad has a working area of 240 × 192mm with a resolution of 320 × 256 pixels. It comes with a free-hand drawing program called Draw, a detachable pen and a choice of three colours (red, green and blue) on a different coloured background.

The Grafpad is available direct from British Micro, Penfold Works, Imperial Way, Watford, Herts.



"It is 'user friendly' but Stanley hasn't learned to talk to it properly yet!"

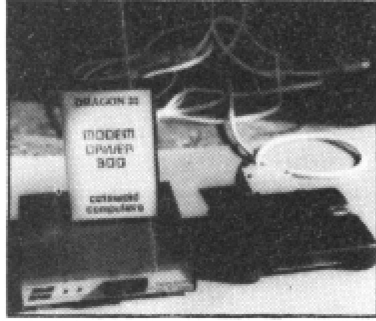


Dial a number

A BB1 Communication package for the Dragon 32 has just been released by Cotswold Computers.

The package allows you to communicate with other Dragon users by means of the telephone system. To do this you need an RS232 serial interface — a device to convert the internal parallel binary numbers used by the Dragon into serial form so that they can be carried on the telephone line; a modem; cables and software.

Cotswold's RS232 interface is a "fully professional unit" which can transmit and receive data, and handle requests to send, clear to send and data carrier detect. Its speed or baud rates range from 50 to 19,200bps, and output from the unit is through a standard 25-way D plug.



The BB1 Communication package

The modem is a British Telecom approved V21,300bps asynchronous hardwired unit which can be positioned on top of the Dragon. Power is by two internal PP3 batteries or an optional mains adaptor and the modem is switch selectable between originate and answer modes. Instruction manuals are provided for both the interface and the modem.

Also included in the system is software to give your Dragon access to over 300 baud systems/billboards/databases and a cable, wired ready for operation.

The system costs £149 plus package and postage and is available from Cotswold Computers, 6 Middle Row, Chipping Norton, Oxon.

Also coming from Cotswold is a cartridge expansion board enabling simultaneous operation of a disk drive and interface, and a disk-based communications package for OS-9. The package (price £25) operates on 110, 300, 600, 2,400, 4,800 and 9,620 baud rates, takes the signal in through the RS232 and creates a disk file which "can be manipulated in any way you like under OS-9" according to Peter Tomlinson of Cotswold.

Making a game

WHAT makes a great computer game? This was the question asked by Microdeal, one of the largest producers of Dragon software, in a survey distributed to dealers throughout Britain.

A genuine challenge was the top answer (chosen by over 30 per cent of respondents) followed by life-like graphics (25 per cent). Dramatic sound effects and a test for quick reflexes tied for third place, and there was good news for readers worried about mounting violence; outlet for aggression ranked bottom of the list.

If you have any thoughts on the essential ingredients of a great game, send your comments in to *Dragon User*.

Price cuts

PRICES are being reduced on a limited number of software titles from Dragon Data, and the company also announced the decision to end large packaging for most of its software.

Shark Treasurer, Whirleybird Run and Galax Attack are down from £12.95 to £7.95; Lunar Rover Patrol, Pixel Editor and Synther 7 are down from £10.95 to £7.95 and Madness and the Minotaur, Calixto Island, Black Fanctum, Quest, Typing Tutor and Dragon Mountain are now retailing at £4.95, reduced from £7.95.

The reductions follow some criticisms of excessive software prices. Steve Davis of Dragon Data said "we had a look at our range and thought that on the above the software was overpriced on the market".

On the packaging front, only disk-based software and cassettes requiring comprehensive manuals will continue to be sold in large packages.

Dragon software on the up and up

***John Scriven's** task is made more enjoyable as the quality of games steadily improves, and new life is breathed into the Dragon's software*

SINCE THAT far-off day in 1982 when I first looked at the tentative offerings for my shining new machine, I must have seen something like 300 programs for the Dragon. I often wish that programmers could have also seen these, so that they didn't try to re-invent the wheel. If there are already 10 perfectly good versions of Alien Splat on the market, producing an eleventh is a waste of time, and changing the shape of the invader hardly counts as original.

Brainwaves

What is needed is an input of new ideas and Dragon owners are lucky that these have at last started to appear. Although there are some long-lasting favourites, a novel program often leaps to the top of the software charts. (Witness the success of Ant Attack and Manic Miner for the Spectrum.)

When I see the umpteenth copy of Invaders I can only feel sorrow for the person who spends their hard-earned cash on something they've already got, sorrow for the firm that has wasted its programming expertise for nothing and sorrow for the stagnant state of the industry.

However — a breath of fresh air has been blowing along this poor reviewer's corridors, and this month a bunch of superb programs arrived on my mat.

Thinking of Spectrum programs like Manic Miner reminded me of one that cost me many hours of sleep when it appeared 18 months ago — **Hungry Horace** from Melbourne House. Perhaps programmers were put off by the Dragon's 6809 processor; whatever the reasons, there have been fewer fast games with interesting graphics than for the Spectrum and Commodore. If Hungry Horace has indeed taken 18 months to convert, then it's been worth the wait.

Your job is to control Horace, an endearing little character who has an awfully large appetite. He runs along paths in a park devouring everything in sight. There are bridges and tunnels, and just to make your task a little more difficult, some park-keepers are out to catch you and throw you out of the park. Once you've reached the end of one section, you can go on to another that is different in shape.

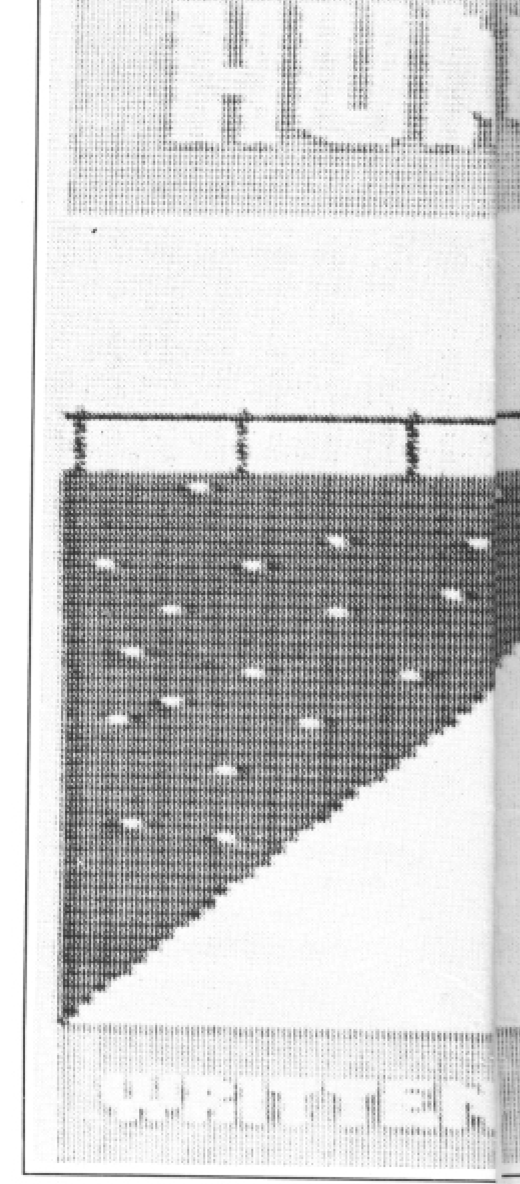
If you steal the alarm bell in any of the sections, the guards rush off in panic, dropping their lunch packs. Naturally enough, Horace will eat these with relish and increase his score. If you're lucky enough to be caught, then you get thrown out ot that section of the park. Control is by means of the cursor keys or a joystick, and the graphics are excellent.

If you like maze chasing with a difference, then I can heartily recommend Horace. Perhaps Microdeal's Cuthbert has a new rival, and soon Horace will be skiing and fighting spiders like he does for Sinclair owners.

Just as Horace is a friendly game, and neither Horace nor the park-keepers get permanently damaged, so **Pedro**, from Imagine, is a mild-mannered program. In fact, the only creatures to suffer are marauding garden pests. Pedro is an unfortunate Mexican who has a beautiful garden full of flowers. He is unfortunate, because everyone from miles around has heard of his prize blooms — including rats, locusts and the village tramp. Pedro can block paths with bricks, chase the invaders and even stamp on them, but they still end up with his plants. He can plant seeds and move compost and bricks, as well as run and jump, but he can only do one thing at a time.

High quality

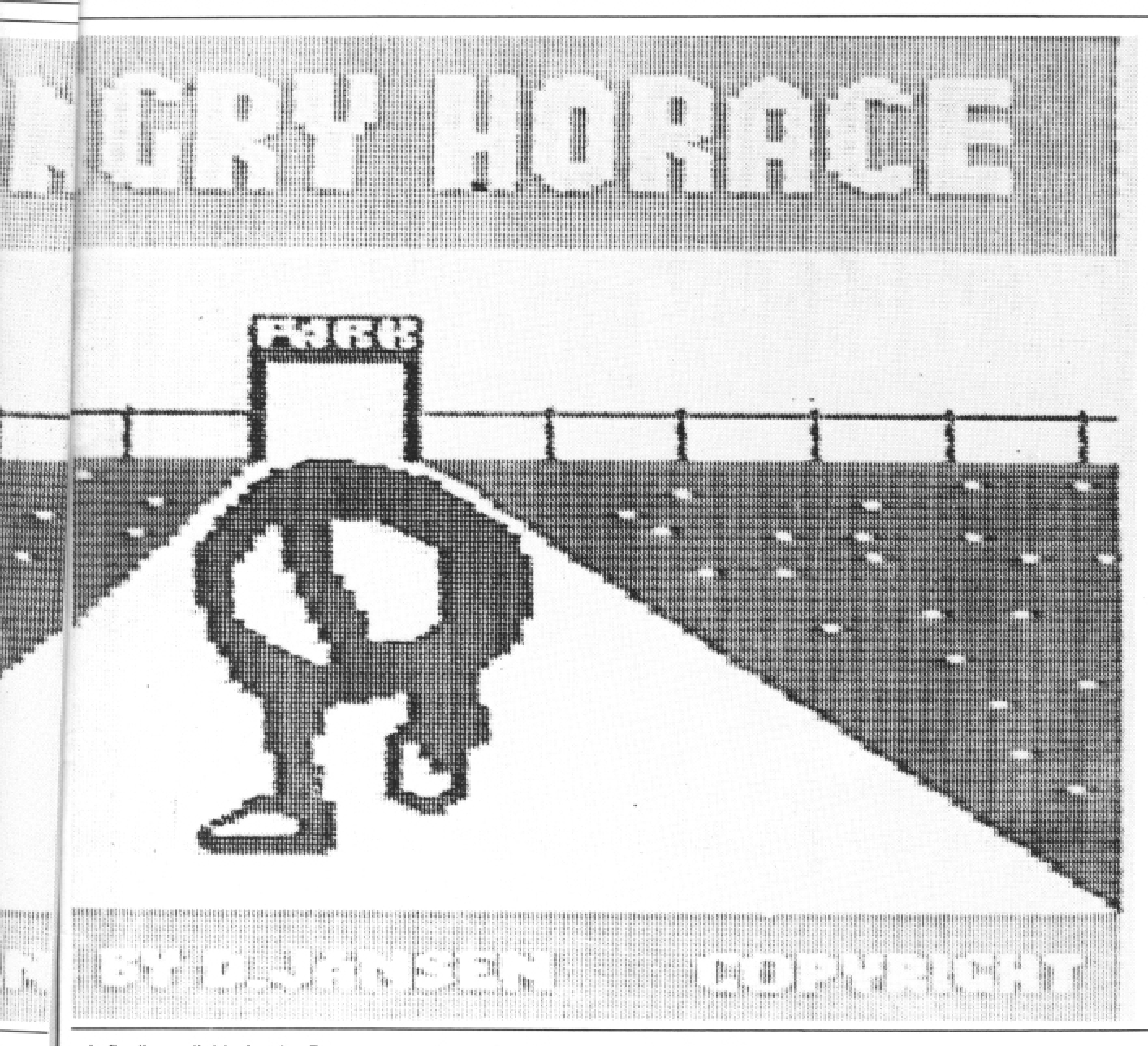
The screen display is good — a 3D view from above and to the side, and has a lot of detail. This is the second Dragon game I've seen from Imagine, and has apparently been released at the same time as versions for other computers. Although the subject may not sound too exciting, Imagine have produced a game of high quality, and it's a pleasant change to see a fine game from a software house on this side of the Atlantic.



After months of waiting, Hungry Horace

Mad Monty from Screenplay is not really in the Pedro class, but is still an interesting game to play. You control a slithering python that needs to devour frogs, toadstools and magic mushrooms that litter a garden. There are different speed levels, and as you consume the various delights, the length of the snake increases, thus making it difficult to avoid crashing into the walls or indeed sections of your ever-lengthening tail. Although it's not a bad game, the slow keyboard scan of the Dragon means that control is not all it could be, and it takes a while to learn the tricky art of snake control.

Another program that involves creatures slithering across your screen is **Ultrapede** from Softek, in fact, a fairly good version of Caterpillar. The Ultrapede rushes down the screen at top speed (at higher levels it resembles an express train). You can move your position by control keys or a joystick, and you have to avoid the Ultrapede while shooting the



orace is finally available for the Dragon — and on page 62 you can win a free Horace game

mushrooms. Destroying the Ultrapede is difficult, as when you shoot it, the separate sections continue to move on their own, and Brian the Snail makes a guest appearance dropping tiny spiders that cannot be destroyed. If you like high-speed garden pests and are a fan of the Magic Roundabout, this could be just what you're after.

One of the most irritating games on offer this week is **Shaft** from Dragon Data. This is nothing to do with the black investigator with a sharp line in leather raincoats, although you can choose to play the part of Ruck Furd or James Bund in the game. The object is to dodge blue lifts that move up and down the screen while moving a small figure across to reach the far side. Here you can take a yellow lift to the next level and attempt to get back to the other side. Gaining points all the time, you eventually reach the top of the screen, however, all is not as it seems, and avoiding the lifts is very difficult.

You may like the challenge of a game as difficult as this one, but it does seem to be pitched towards the top of the difficulty tree, and there seem to be small rewards for all the frustration.

Watch out for Terry the Pterodacty! in Ugh!

About three years ago, someone in a dark corner of Japan who clearly enjoyed watching old American movies decided to write an arcade game about a giant gorilla which had captured a girl. Although computer graphics aren't up to portraying the assets of Fay Wray (or Jessica Lange, for that matter) to their full, the game took off. Unfortunately, the programmer was more used to Kanji characters than western letters, and instead of calling it Monkey Kong, he called it Donkey Kong. Thus was born the entire generation of barrel dodging figures clambering up the scaffolding to the top of the Empire State or Twin Trade Towers in Manhattan to rescue the damsel in distress. This game has been criticised by feminists as being typical male chauvinist fantasy. To counter this claim, you might have expected a version whereby a woman is portrayed in the role of rescuer. Instead, what emerged was a game in which the gorilla liberation front sent in their best fighter, none other than Kong's son, Junior.

Dragon Data calls its version **Junior's Revenge**, and it is a very good copy of the arcade game. Luigi has captured your father, King Kong, and you have to rescue him from his cage. Normally a simple task, this is made more complicated by the trained animals Luigi employs to frustrate your task.

Captive

The first screen is comparatively simple, and consists of various islands and vines that enable you to reach the cage where your daddy is held captive. Swinging across with your joystick, you can reach the second screen which involves pushing keys into place while climbing up long chains. The third screen is like the first, but the fourth has conveyor belts and trampolines to confuse you, and after some repetition you will eventually arrive at the killer — screen seven, Luigi's hideout. This has both conveyor belts and sparks to avoid, and you may eventually be allowed to get close to releasing your father.

This is an excellent copy of a good arcade game, and has a choice of practice or real games. A difficult game to master, this is one that should be included in any serious collection.

The next game I looked at would have sold well two years ago, when a little island in the South Atlantic absorbed most of our attention. In many ways, I'm glad this game has only just appeared, as modern warfare and video games have approached each other to the level where they are sometimes difficult to tell apart. **Jump Jet** sits you at the controls of a Harrier fighter. You control the movements by means of the joystick, and are instructed to fly the plane through a wave of enemy bombers attempting to knock out the airstrip to an island▶

◀where you will see the enemy fortress. There are 10 levels, ranging from difficult to merely impossible, and the game includes speech synthesis at a rather rudimentary level. "Get lost" it kept muttering at me until I realised it was saying "jet lost". It also rattles out "attack" through your tv speaker with something approaching the clarity of an airfield tannoy system.

Flying high

In spite of my earlier reservations, this is an exciting game, although the wrap-around screen is irritating: when you fly off the right-hand side of the screen, you reappear on the left side, and the display scrolls so slowly that the effect can be rather disturbing. Certainly it will take a lot of skill to fly to the island and destroy the enemy.

As you start with eight planes you could imagine that your task would not be too hard. In fact, the attacking planes are capable of bombing the runway and destroying remaining planes, so you end up with less machines than appear at first sight. The difficulty is set by your ability to succeed at each level, although the initial level can be set at the keyboard. Pressing the reset button allows you to choose the initial level again. (Level nine gives you two planes on the runway, but you don't stand much chance of even leaving it.) The animation could be smoother, but the game is entertaining and fairly novel.

Although it brings back sad memories of recent naval campaigns, if you can forget the connotations, you may find this game a suitable addition to your collection.

My arcade-action award this month has to go to Hewson Consultants for **3D Space Wars**. Last year, they issued a tape called Dragonfly, which was a competent, if rather elementary flight simulation program. 3D Wars is also a flight simulation program, but one that has indeed come of age. This program gives you the view from the cockpit of a spacefighter set against a fleet of Sieddab raiders. Your mission is to destroy them while maintaining a reasonable level of fuel.

The screen is full of the enemy space ships swooping and diving at you in a most realistic manner. All the while you are firing at them, they are shooting back, and your fuel reserves are getting low. The solution to this problem lies in locating a refuelling vessel that you must line up in your sights. Woe betide those of you foolish enough to blast it with your lasers! The whole secret to this game lies in locating the refuelling vessel every four minutes or so and making good use of it.

Without refuelling, your task (whatever it may be) is doomed to failure. The screen display is superb, with enemy ships zooming at you and increasing in size as they approach, as well as shell-bursts on the nose of your ship and laser shots that are shown graphically. Although the combined forces of the baddies, oops, Sieddab raiders, is always more than mortal man can face, Hewson Consultants have produced an excellent game that forces you to send the rest of the family away while you turn up the tv volume and lose yourself in inter-galactic battles. Definitely my favourite game of the month.

Any firm that calls a game **Ugh!** can't be all bad, and Softek have recently unveiled a program that goes by this inauspicious title. Although early man had merely to contend with mammoths and sabre-tooths, like the film 1,000,000 years BC, this game gives you the opportunity your forefathers never had — being chased by Pterodactyls and T Rex's while pinching eggs from the former.

Watch out

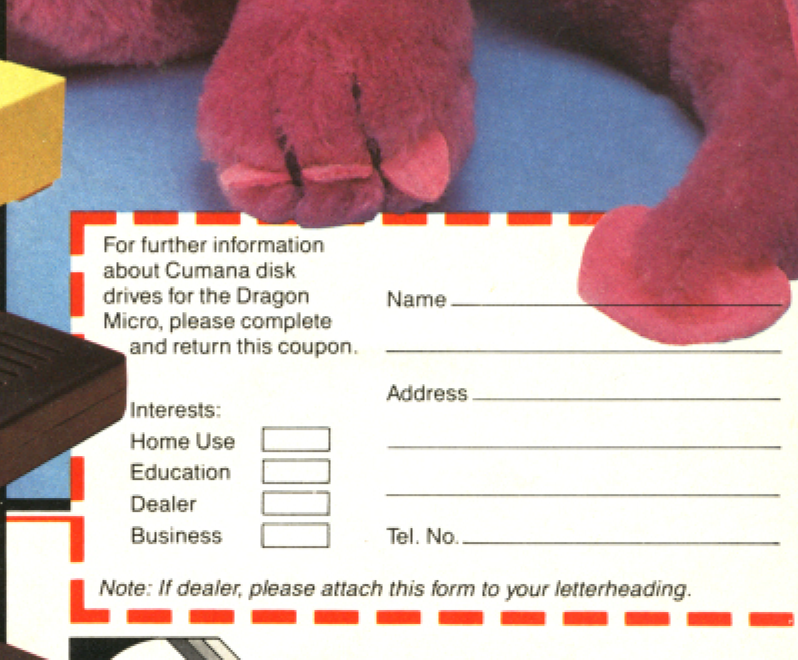
The screen shows a couple of zig-zag paths with three caves. You control the hero, called Ugh! and steer him to the pile of eggs. On the way you may meet Rex, the Tyrannosaurus, but if you're not carrying any eggs you can throw a spear at him. While this is going on, Terry the Pterodactyl is dropping rocks on to your head. You have four lives, and there is only one space on the screen that is safe — your home cave. As you spend more time escaping from Rex, you forget to look up in the air, and an awful thunk sound comes from the speaker. This is a novel game that is fun to play, and there are comprehensive instructions at the start. There are several different screens and 18 skill levels that change▶

BUY YOUR DRAGON A FRIEND

Here's a friend your Dragon has always wanted — a Cumana disk drive; and Cumana slimline disk drives are friendly to you, the user, as well as to your pocket.

Designed and manufactured to the highest standards, Cumana disk drives have an independent power supply, 12 months warranty, and are fully assembled and tested before packaging. As part of the package, your first disk drive for the Dragon — addressed by Cumana as drive A — is supplied with a comprehensive user manual, 'DELTA' ROM based cartridge adaptor and demonstration diskette. Upgrading your system is simple, and up to four Cumana disk drives can be added without any modification to your microcomputer.

Cumana slimline disk drives for the Dragon Microcomputer are now available from Spectrum UK, as well as area distributors and Cumana's national dealer network. Look out for the distinctive Cumana packaging in your high street, today!



Dragon is the registered trademark of Dragon Data Ltd.
Delta is supplied by Premier Microsystems Ltd.

CUMANA
The best name in memory

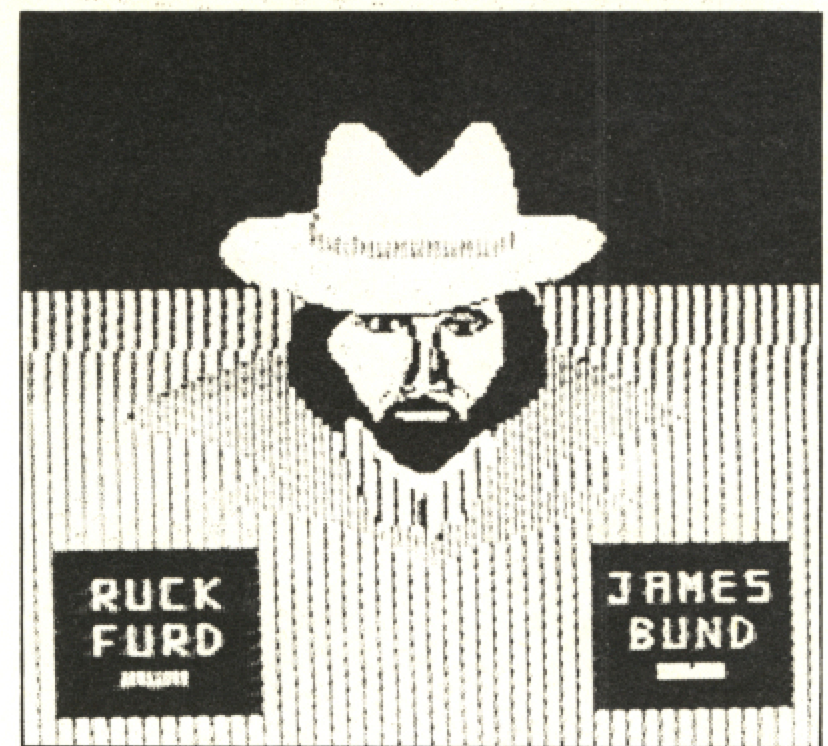
Cumana Limited, Pines Trading Estate,
Broad Street, Guildford, Surrey, GU3 3BH.
Telephone: Guildford (0483) 503121 Telex: 859380

◀immediately as you improve, so you will be unlikely to tire of Ugh! too quickly.

Also from Softek is **Galacticans**, so you can guess which arcade game this is supposed to represent! Hordes of whirling ships descend from the top of the screen, while you race left and right along the bottom avoiding them and frantically firing your laser. When you clear one screen, you are confronted by . . . just another screen with more of the unpleasant creatures.

There is on-screen scoring, rather nice sound effects, and the display is reasonably clear. Why then should I sound as if I have my doubts? I enjoy well-written fast space games, but this seems almost as boring as I*V*D*RS. There doesn't seem to be much point in playing again, and it's not likely to be the kind of game that involves many different skills. The sort of space game that I'd pay money to play in an arcade would have to be something of the calibre of Moon Cresta or Defender, and however well Galacticans may be written, it looks a little old-fashioned set against the latest Dragon games.

Lothlorien have been producing war games for the Dragon for some time now, usually set in the past. **Johnny Reb** is no exception, and as the name implies, is set during the time of the American Civil War. You can use the computer merely as a display screen and play against another human, or you can pit your wits against the computer itself. There are several options available at the start, including whether you wish to fight for the Yankees or the Rebs, and how you wish to divide up your army into artillery, cavalry and infantry.



Play James Bund or Ruck Furd in Shaft

When play starts, you are shown the playing area, with a river running down the middle. Your troops are arrayed on one side, and the computer's on the other. You can move your pieces around the screen as well as firing the artillery pieces, and the game is over either when one side's flag is captured or when a pre-selected time limit has expired.

Fire

Having seen previous Lothlorien games, I was expecting some exciting graphics during the interchanges, but apart from the odd cannon-ball whizzing across the screen, nothing much happens, and the responses at higher levels are rather slow. If you are a serious war-gamer you will probably find this program entertaining, but it didn't interest me as much as Tyrant of Athens and Samurai Warrior.

Shards is well-known for its series of educational and adventure programs. This month I've been looking at something completely different from them. **Shaper** is a sound utility program that allows you to define your own sound effects and store them for use in your own program. The package contains the main program together with a library of pre-programmed sounds. As the Dragon does not have its own sound chip, some manufacturers have produced add-on devices. The problem (pointed out in Shaper's documentation) is that programs written to utilise these will only work on other Dragons that also have the sound board.

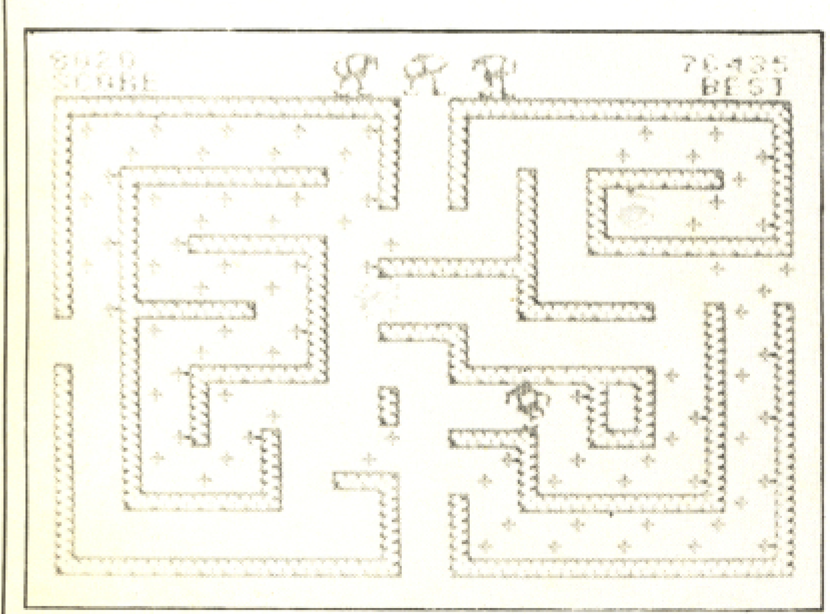
Based on a sound idea (sorry!) how does the program perform in use? Firstly, it's simple to program your own sounds. After the rather noisy opening screens, the initial menu appears, giving you the choice of explanations, assistance, listening to the library and building your own sounds. If you select "build" as an option, you will be given instructions and then shown a list of the 10 parameters that have to be entered. These include squeeze, expand, and chain, as well as the more usual volume, frequency and envelope values. At any time you can press the space bar to listen to the sound so far.

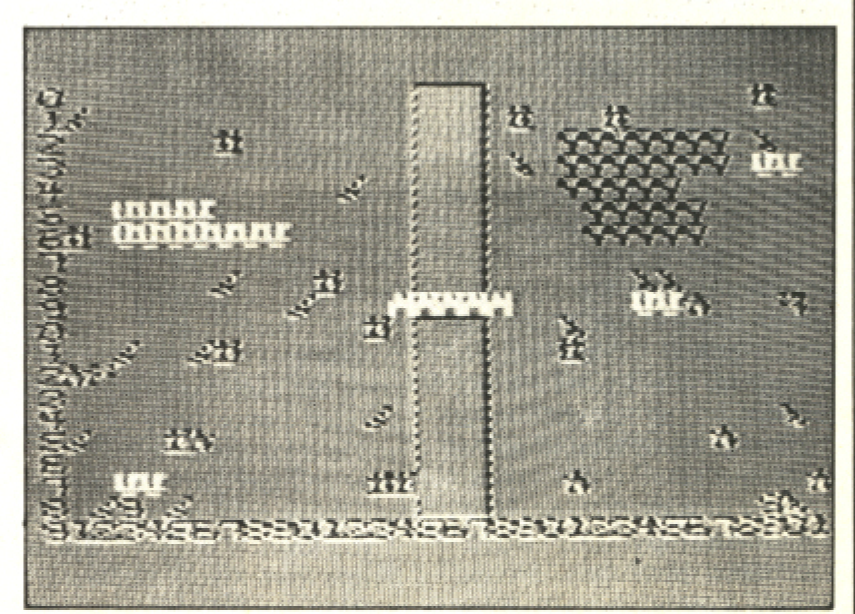
Select sound

When the sound is to your satisfaction, you can select the save option and store the sound effect on cassette. You can now enter the program you wish to employ your sound in, and load in the effects at the end. If you prefer, you could use one of the sounds included in the library. Some of these are quite amazing, and vary from UFOs and tank battles to fairly melodious synthesiser sounds. At the end, you are shown how to use the start of the Basic program to load in graphics and title pages with your own program.

Accompanying the program is an A6 booklet (a quarter the size of this page), with 36 pages of detailed notes and diagrams. This is very well written and makes using the program simple. As it is said: if all else fails, read the manual. In this case, the program is well documented internally, but the booklet adds the finishing touch to a very fine utility.

When the Dragon first appeared, I'm sure that many people bought it because of its "real" keyboard. They probably imagined that it would be useful as a word-processor, and as a text storage medium.


Maze chasing with a difference in Hungry Horace


The opposing forces in Johnny Reb

Unfortunately, when they arrived home with their new toy, they were rather disappointed to discover that it didn't have lower case on the screen, and the display was only 32 characters wide. Word-processing programs have appeared that tried to cope with these problems by redefining the character set and reformatting the screen. Mostly, however, they are expensive — like the Microdeal version.

There are some programs about that put up with the Dragon's limitations and attempt to use the computer as it is. One such program came my way this month, **Editext**, from Nectarine. In fact it's really two programs, as there is a disk version included on the cassette at no extra cost.

Options

On loading, the initial display is a menu with nine options: new text, view lines, edit, continue, print, save, load, line editor and end. The text is entered in one large chunk, and the program takes care of word-wrap, so you don't have parts of words hanging over ends of lines. If you view the text, you discover that each line has been given a number, and this makes line editing easier. The program can store up to 450 screen lines, which is somewhere about six sides of A4 print-out. Most of the menu options are fairly self-explanatory, and there is an A6 booklet containing 18 pages of clear notes.

There is a separate printer menu giving the options of draft or formatted print, and the opportunity to change printer format and typeface. The formatting commands include all the usual ones of left and right margins, line spacing and number of copies. You can also choose to have the text justified (spread out so that the words touch both left and right margins, as most magazines and newspapers are arranged). If you choose to go to the typeface menu, then you can change the size and design of the letters according to your printer. On loading, it is set up for the Epson FX-80, probably one of the most popular printers around at the moment.

I've usually been disappointed by the word-processor packages around for the Dragon, and the only good ones seem overpriced.

It's clearly unfair to compare Dragon programs with those designed for a machine with 80-column display; however, Editext is easy to use and reasonably cheap. If you can accept the disability of never seeing your formatted text until it issues forth from the printer, then you could do worse than plump for this program.

Hungry Horace £5.95	Melbourne House, Castle Yard House, Castle Yard, Richmond TW10 6TF
Pedro £5.50	Imagine Software, Tithebarn House, Tithebarn Street, Liverpool L2 29P
Shaft, **Jump Jet** £7.95 each; **Junior's Revenge** £10.95	Dragon Data, Kenfig Industrial Estate, Margam, Port Talbot
3-D Space Wars £7.95	Hewson Consultants, 48 Grand Parade, Brighton, East Sussex
Ultrapede, **Ugh!**, **Galacticans** £6.95 each	Softek, 12/13 Henrietta Street, Covent Garden, London WC2E 8LH
Johnny Reb £6.95	Lothlorien, 56a Park Lane, Poynton, Cheshire
Shaper £9.95	Shards Software, 189 Eton Road, Ilford, Essex
Editext	Nectarine, n/a
Sprite Magic £17.25	Knight Software, 93 High Street, Eston, Middlesbrough, Cleveland
Mad Monty £7.50	Screenplay, 134 St Vincent Street, Glasgow G2 5DU

Friends of yours who own Commodore 64s probably boast about their wonderful facilities, including upper/lower case and sprite graphics. I'm sure that you point out that the Dragon has a much better copy of Basic and is a lot easier to program. Now you too can have sprites and lower case on the Dragon if you own a copy of **Sprite Magic** from Knight Software.

With sound utilities, you may possibly get better effects with plug-in cartridges, but the programs you write will only work on machines that also have the cartridge. The same is true of graphic enhancers. If you use a purely software-based utility, you can transfer the programs to any Dragon. The main program consists of some machine code that is loaded at the top of memory, and acts by adding extra key words to Basic. Once this is loaded, you can write a normal program to use the new words or look at the demonstration programs.

There are over 40 new statements that are very comprehensive and allow you far more than simply defining up to 128 different sprites and moving them around the screen. There is collision detection, movement control by keyboard or joystick, and maze modes where sprite zero chases all the others or vice versa. There are a whole set of new error messages, enhanced sound commands and even a scoring feature, so games are very easy to implement. These are described in detail in the 24-page booklet that comes with the package. The set of commands are more comprehensive than in any other utility I've seen, and include more than you get on, say, Simons' Basic for the Commodore.

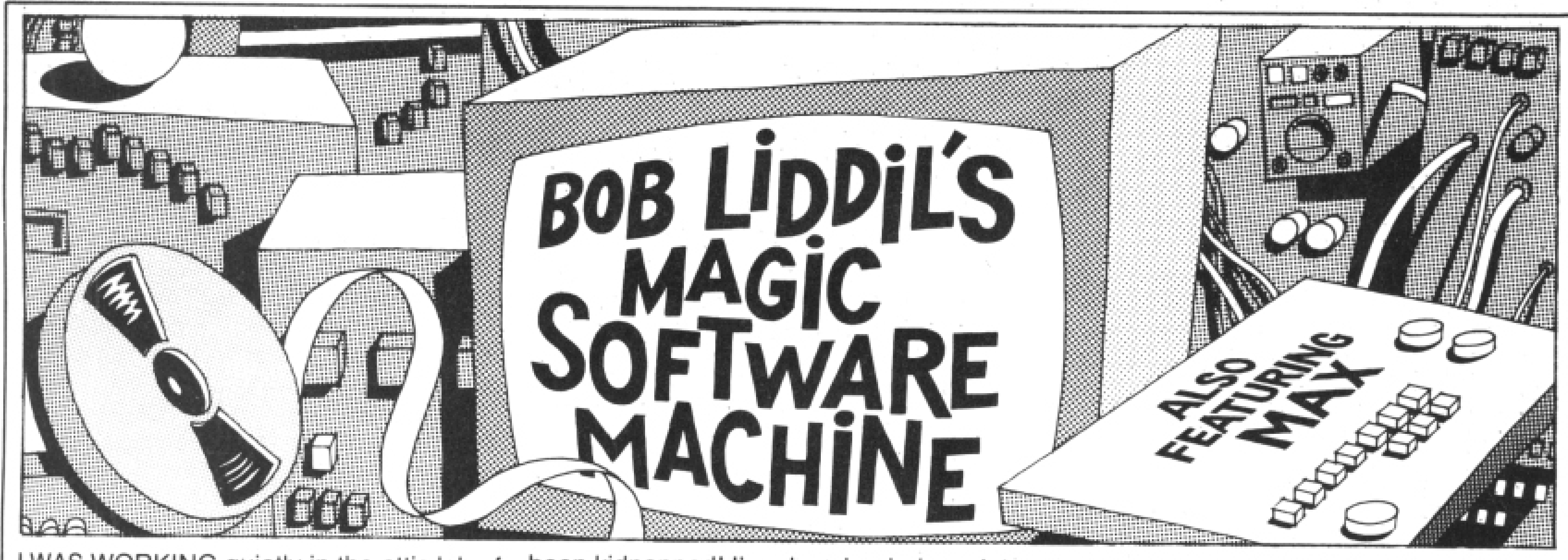
One of the nice features of this package is the collection of demonstration programs. The first simply shows large characters floating around the display. When you press break, you discover that the whole character set has been redefined four times as large, so pressing List looks rather strange. The next program gives you normal sized upper/lower case as well as the opportunity to redefine all the character set using cursor control keys on a large grid.

Program three lets you output new sounds, and number four shows a large chess board with moving pieces that execute "fool's mate". The winning pieces even jump up and down after their victory! Program five lets you shoot at some fairground ducks that drift across the screen, and number six is a version of Breakout, where you attempt to demolish a wall. The final program is a maze chess game that demonstrates sprite control. This is a very professional program and can certainly be recommended.

Coming soon

Next month I hope to look at some more sprite utilities as well as the latest games, two language packs — Pascal and Forth — and also a compiler to add some zip to your Basic programs.

I find it hard to believe that for two months now I've been faced with such good and varied software. Instead of being the poorer relation in the family of micro owners, compared to their Sinclair and Commodore cousins, Dragon owners now have as large a choice of quality software as anyone else. The problem must be which programs they can live without. There are many excellent programs on the market and their standard is improving all the time. Let's hope it continues to do so. ■



I WAS WORKING quietly in the attic lab of Professor Megabyte's old Victorian house when I got a call from the Chief on my two-way wrist modem and pocket computer. An anonymous tipster had revealed a clue to the whereabouts of that missing academic and genius-whiz hardware designer, for whom I had been searching lo these many months.

On my desk was a British manufactured tape called Maths Trek, from Dragon Dungeon's Dungeon Software line of programs. Max had converted it to disk for me. According to the Chief, somewhere in this software would be everything I would need to find the long-missing Professor.

Since this program is designed for the Dragon 32/64 (which I just happen to have a have a Tano version of), I knew I would be able to get to work right away.

I really didn't know what to expect when I booted up Maths Trek. Max wasn't there to brief me on the details. He's been wandering the phone lines more and more, visiting data bases in strange places like Sandusky in Ohio, and Hull in England. Then it hit me — that sinking feeling I get whenever I am downloaded into the world of a computer program.

"Max — what are you doing?" I shouted in vain as I re-rezzed in real time. There was no answer. He wasn't anywhere within earshot and I knew I was in for a rough time.

I re-rezzed in the padded seat of a Klingon Class VII light attack cruiser, inside Maths Trek. Directly in my sights was the awesome, formidable heavy starship USS Enterprise, phasers to bear, photon torpedoes armed, shields up and primed for battle. All this hardware was sighted on *me*, according to my special patented Likelihood of Destruction Calculator and digital watch (which I never leave home without).

This universe was more bizarre than any I had ever visited in all the time I've known Max. Instead of asteroids in the normal sense, this space was littered with sculptured planetoids shaped like maths problems. There were rock numbers shaped into equations of addition and subtraction, multiplication and division. No doubt about it. Trouble had caught up with me.

It was at exactly this moment, when I believed matters could be no worse, that I heard a voice over the radio. It was Max.

"Bob, don't transmit, just listen. I've been kidnapped! I've downloaded you into the first program booted and encrypted sideslip codes to access my prison. The menace you face is greater than even that of cancellation of your column. Help me, Bob, the fate of the world is in your hands."

I didn't have time to respond or absorb the reality of the situation. Somewhere, someone pressed enter and the game was on.

Maths Trek is a Basic language educational game from Pete Wood at Dragon Dungeon in Derbyshire. It combines the graphics delights of an arcade/illustrated simulation with the functionality of a rote mathmatics drill.

The USS Enterprise is powered not by dilithium crystals, as has been the case in other "trek" style programs, but by the answers to maths questions posed within the context of user-defined skill levels. An input of 42 to the question "6x7" would increase speed, repair damage, destroy enemies, and (not incidentally) remarkably increase the ability of the player to do his/her maths.

Pete Woods is to be congratulated on his courage as it is a well known fact that everybody wants educational programs until somebody offers one. Then they gather dust in favour of arcade or other gamefare.

Battle royale

Dungeon Software has created a marvellous, well organised unit of software. The presentation is superb, the intent (which is to teach maths), is well preserved. The packaging is pleasing to the eye, which should in turn please distributors and dealers. The end user will find, as did the father of a little lad in Wakefield, that schoolkids can't get enough of it. (I witnessed the battle royal to put these tykes to bed while they were still playing.)

Maths Trek fulfils all its obligations, as a reasonably priced competitor to what little other educational Dragonware is available. It is a sturdy program that everyone with kids should have.

Often when I am downloaded into Software Worlds I become an equation in the scenario. Rarely do I emerge on the antagonistic side (although I was once an invader in a Space Invaders game for the TRS-80 Model 1). Still, when faced with being scrooched by enough phaser power to reduce Yorkshire to pudding, one must use his wits or be vaporised. I opened a hailing frequency.

Too late! A scathing bolt of pure phaser plasma danced against my screens (which fortunately held), severely jarring the interior of my cheaply made ship and spilling Klingon tea all over the deck. The acrid fumes of dissolving metal filled the pilot's cabin as the tea began eating through the floorplates. No wonder those Klingons are on everybody's case all the time.

"Jim, quick!" I rasped through the microphone, "beam me aboard!"

The surprise of hearing his first name jolted the Enterprise captain into action. I was beamed from my semi-disabled cruiser and whisked into the transporter room where a very alert Mr Scott nearly dropped his hand stunner in surprise to find a human (me), not a Klingon materialising.

The doors to the transporter room whished open and in stepped Captain Kirk followed by a short, pudgy, balding fellow dressed in a lab coat. It wasn't Dr McCoy or anybody that should have been here.

Then it dawned on me. I smiled, took a step forward and extended my hand.

"Professor Megabyte, I presume," I said. He seemed startled that I knew him. "Sir," I said, "we certainly have a lot to discuss.'

In the crew's lounge, several pints later, having briefed Professor Megabyte on situations past and present, it was decided that action should be taken to rescue Max. The Professor, using Federation Technology, had devised a Manual Sideslip Activator, against the day when power failure or other disasters should befall this universe.

After bidding farewell to Captain Kirk and the crew of the Enterprise, the Professor set off the signal that would lead us to Max's prison . . . and his jailers.

In realworld a disk drive whirled and an encrypted alogogram activated the forces of sideslip. The Professor and I de-rezzed on the bridge of the starship and plunged dizzily straight down, through the depths of the very core of software time and space. We fell at a dizzying rate, plummeting through planets and stars, through invaders and defenders. Downward and downward we spun until I thought I would surely be ill. For a time there was no sensation of motion at all. Then we slammed helter skelter into a darkness to end all darkness. We'd collided with a black hole. ▶

◀ From now on, I sleep with a night light. When the world came back into focus I was staring up at a deep red sky. Someone stood over me, prodding me, enjoining me to wake up, which I did with a start.

"We must get out of sight quickly," the incredibly beautiful blonde in the shimmering silver and rhinestone jump suit said. So saying, she escorted the Professor and myself to what she described as Resistance headquarters.

"Now let me get this straight," I said, trying not to be distracted by her unbelievable beauty, "this fellow Twitch has recruited imperfect video games and lost subroutines from the outer reaches of RAM, overthrown the legitimate government of the CPU, captured an extremely powerful out-of-system program which he intends to harness to help him capture some heretofore unseen/unknown world. Is that correct?"

She nodded. "And so far no one has been able to stop him?" Again the nod.

Hmmm, so that's why only arcade games have been loading on my Dragon lately (apparently the Maths Trek qualifies due to its extensive use of graphics).

It was agreed that Professor Megabyte would seek help while I and my lovely new associate would try and spring Max quietly.

That scheme quickly turned sour when we were discovered by a large aggregation of nasty-looking palace guards recruited from the program Phantom Slayer. On our way to the nick I noticed a helicopter from Chopper Command parked on a grid outside the main CPU palace. Obviously Twitch was gathering stuff from other than defective programs. That chopper is the baddest of all Gamedom.

We were kept in custody for a short time and then escorted into the main throne room. It was furnished in Fortress of Ultimate Darkness Modern. Any second I expected the Supreme Being to pop in hollering, "Bring back what you have stolen"! And this guy Twitch looked like the villain of every bad movie I ever saw.

Sinister sneer

He glared evilly at me. "My plan has succeeded!" he sneered, "now I have the program and his master all in one neat little package." He gestured to some minions in the shadows. "Bring in the outworld program!" he bellowed.

They dragged Max in and dashed him to the floor. He was injured but not seriously. When he saw me his face fell. He must have believed at that moment that all was lost.

Just then a mighty explosion rocked the palace. Then a second, and a third. A transporter beam separated into five shafts of light, materialising Captain Kirk, Mr Spock, Mr Sulu, Mr Chekov and Professor Megabyte.

Faster than you could say "Nanosecond", Twitch and his personal guard had vanished through secret doors behind the throne. At the same time, a well co-ordinated rebel attack eliminated any resistance beyond the gates in CPU City.

It was all over. Stability had been restored, Max had been rescued and Professor Megabyte had been found. After a brief round of handshaking and congratulations, the Professor dialled in some codes. The unmistakable wrench of upload gripped us. In a shimmer of dissolving binary codings, CPU City dissolved.

When everything came back into focus I was back in the lab, alone. Well, not quite alone. Max was back in his box. The lights in his expando flashed enthusiastically. But Professor Megabyte was nowhere to be seen.

It was then that I noticed that Maths Trek was still on line on my Dragon. It was waiting for the answer to a maths question on the screen.

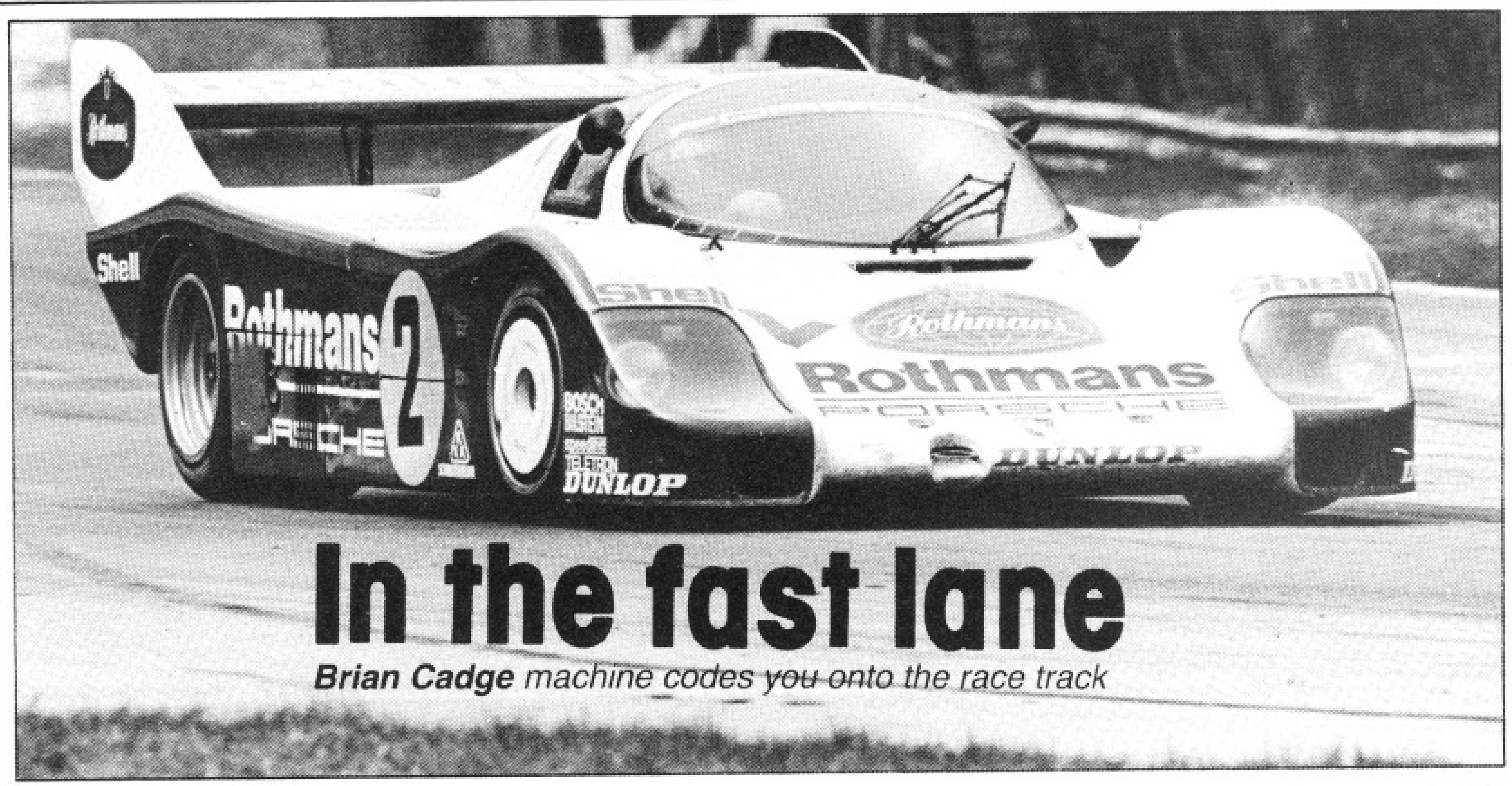
I typed in the numbers 42 and pressed ENTER. The screen changed to a colourful view of the galaxy and the Enterprise picked up three warp factors. Down at the bottom of the display appeared a short message. I read it and started to laugh.

It said: Thanks for the boost and best of luck. It was signed, Professor Megabyte.

That was it then. The Professor preferred the world of the Trek program to Realworld. He had always been a trekkie anyhow, so the fact of his remaining aboard the Enterprise seemed reasonable.

As for Max and me, this being our last appearance in *Dragon User* magazine, we will simply bid you farewell. Britain and the Dragon will hold a special place in my heart.

Goodbye Britain, and God bless. ■



In the fast lane

***Brian Cadge** machine codes you onto the race track*

GRANDPRIX IS A machine code game for one or two players using joysticks. Player one uses the right joystick and player two the left. When RUN the program will ask for the number of players — enter 1 for a single player game, 2 for a two-player game.

The program, which is about 1K long, uses PMODE3 hi-res colour graphics and realistic sound. The object of the game is to achieve the highest score in the three minutes that a race lasts. If you hit one of the other cars you will lose one of your three lives. The joystick controls left and right movement of your car, while pushing the stick up will cause you to speed-up and pulling it down slows you down. The sound of the engine indicates speed.

Being written entirely in machine code, the game is very fast; also very realistic engine and crash sounds can be generated continuously at the same time as the graphics are animated. The score is constantly shown and the faster you dare go the faster your score will rise.

I have included two programs to enter and check the code. Once entered, save the program *before* running it. To do this type: CSAVEM "CARS", 28000,29183,28732. Type EXEC 28732 to start the game. Note: before typing in or loading you must type PCLEAR6 to reserve the extra graphics RAM needed. If the game does not work, enter program 2 and check the data against the listing. Location 28001 contains the number of lives per game, pole this with any value 1-255 to alter the difficulty.

As several ROM routines are used, the program will not run on a Tandy colour computer.

Happy racing . . .

```
10 'PROGRAM TO ENTER MACHINE CODE
20 PCLEAR6:CLEAR200,27999
30 CLS:B=28000
40 PRINT B;:LINE INPUT A$:IF A$="" THEN 40
50 IF A$="END" THEN STOP
60 Z=VAL("&H"+LEFT$(A$,2)):POKE B,Z:B=B+1
70 A$=MID$(A$,3):IF A$<>"" THEN 60 ELSE 40
```

Program 1: entering machine code.

```
10 'PROGRAM TO CHECK MACHINE CODE
20 CLS:INPUT "START & END ADDRESS";S,E
30 FOR I=S TO E STEP 8
40 PRINTI;
50 FOR J=0 TO 7
60 H$=HEX$(PEEK(I+J)):IF LEN(H$)<2 THEN H$="0"+H$
70 PRINT USING " %%";H$;
80 NEXTJ:PRINT:NEXTI
```

Program 2: checking machine code.

GRANDPRIX GAME MACHINE CODE LISTING:

START BY EXEC 28732

continued on page 22

```
28000 86 03 B7 01 3D 7F 71 78 7F 71 79 7F 71 7A 7F 71 7B 7F 71 7C 86 FA B7 01
28024 37 7F 01 13 8E 23 28 BF 00 8D 8E 09 0F BF 01 40 86 08 B7 01 36 7F 01 3A
28048 8E 0E 0A BF 01 42 8E 14 0B BF 01 44 8E 1A 0E BF 01 46 86 E5 B7 FF 22 B7
28072 FF C0 B7 FF C3 B7 FF C5 B7 FF C8 B7 FF CB 86 FF 8E 08 00 A7 80 8C 20 00
28096 25 F9 8E 08 00 86 AA C6 0A A7 85 5C C1 16 25 F9 30 88 20 8C 20 00 25 EF
28120 BD 6F 59 BD 71 2B B6 01 13 81 02 25 0A BD 6E 29 7D 01 3A 10 26 01 11 8D
28144 2A 10 8E 01 40 C6 04 AE A4 BD 6E B7 30 88 20 8C 20 20 25 08 8E 05 C0 BD
28168 6F 3F 30 86 AF A4 31 22 5A 26 E4 BE 00 8D 26 C0 16 02 6C B6 01 37 4C C6
28192 0A 3D 1F 01 30 1F 26 FC 39 34 36 7F 01 13 BD BD 52 7D 01 39 27 0C B6 01
28216 5C B7 01 5A B6 01 5D B7 01 5B B6 01 5A 81 0A 22 0A 7D 01 36 27 05 7A 01
28240 36 20 0E 81 35 25 0A B6 01 36 81 09 27 03 7C 01 36 8E 19 8A B6 01 36 30
28264 86 4D 27 10 86 AA C6 0E 34 10 A7 1F 30 88 20 5A 26 F8 35 10 B6 01 36 81
28288 09 27 10 86 AA C6 0E 34 10 A7 03 30 88 20 5A 26 F8 35 10 8D 3E BD 6E B7
28312 B6 01 5B 81 0A 22 08 7D 01 37 27 03 7A 01 37 81 35 25 0A B6 01 37 81 FF
28336 27 03 7C 01 37 35 B6 34 36 C6 0F 10 8E 71 4B A6 A0 A7 84 A6 A0 A7 01 A6
28360 A0 A7 02 30 88 20 5A 26 EE 35 B6 A6 84 81 AA 27 04 B7 01 3A 39 A6 02 81
28384 AA 27 04 B7 01 3A 39 A6 89 01 E0 81 AA 27 04 B7 01 3A 39 A6 89 01 E2 81
28408 AA 27 04 B7 01 3A 39 39 34 12 86 3F B7 FF 23 7F 00 B2 8E B7 98 A6 80 8D
```

```
28432 22 A6 80 8D 1E 7A 00 B2 26 F3 86 37 B7 FF 23 35 12 7A 01 3D 10 27 01 5F
28456 86 FA B7 01 37 7F 01 13 16 FE 4F B4 00 B2 B7 FF 20 86 C8 4A 26 FD 39 34
28480 34 BE 01 3B 30 03 BF 01 3B 1F 10 3D 3D 81 09 23 04 80 09 20 F8 8B 0A 35
28504 B4 34 36 8E 71 7C A6 84 4C A7 84 81 0A 25 09 4F 8D 15 6F 84 30 1F 20 EE
28528 A6 84 8D 0B 8C 71 78 27 04 30 1F 20 F3 35 B6 34 36 C6 0C 3D C3 71 87 34
28552 06 31 89 8E 88 1F 20 86 02 3D 10 8E 09 40 31 A5 35 06 1F 01 C6 06 A6 80
28576 A7 A4 A6 80 A7 21 31 A8 20 5A 26 F2 35 B6 20 20 20 20 20 20 57 45 4C 43
28600 4F 4D 45 20 54 4F 20 47 52 41 4E 44 50 52 49 58 0D 0D 0D 0D 20 20 4E 55
28624 4D 42 45 52 20 4F 46 20 50 4C 41 59 45 52 53 20 28 31 20 4F 52 20 32 29
28648 20 3F 00 20 20 20 20 20 20 20 20 20 20 20 20 50 4C 41 59 45 52 20 00 20
28672 20 20 20 20 20 20 20 47 41 4D 45 4F 56 45 52 20 50 4C 41 59 45 52 20 00
28696 20 20 20 20 20 41 4E 4F 54 48 45 52 20 47 41 4D 45 20 28 59 2F 4E 29 20
28720 3F 3F 00 53 43 4F 52 45 53 3A 0D 00 BD BA 77 8E 04 80 9F 88 8E 71 7D 6F
28744 80 8C 71 87 25 F9 8E 6F AD BD 90 E5 7F 01 39 BD BB E5 81 31 27 0B 81 32
28768 26 F5 86 01 B7 01 38 20 03 7F 01 38 BD BA 77 8E 05 00 9F 88 8E 6F EA BD
28792 90 E5 B6 01 39 8B 31 BD B5 4A 8D 69 7E 6D 60 BD BA 77 8E 05 00 9F 88 8E
28816 6F FE BD 90 E5 B6 01 39 8B 31 BD B5 4A 8E 04 00 9F 88 8E 71 78 8D 56 8D
28840 64 8D 42 B6 01 38 B1 01 39 27 05 7C 01 39 20 B4 BD BA 77 8E 04 40 9F 88
28864 8E 70 32 BD 90 E5 8E 71 7D 8D 32 BD 90 A1 8E 71 82 8D 2A BD 90 A1 BD 90
28888 A1 8E 70 17 BD 90 E5 BD BB E5 81 59 10 27 FF 54 81 4E 26 F3 39 34 12 86
28912 03 8E FF FF 30 1F 26 FC 4A 26 F6 35 92 34 36 C6 05 A6 80 8B 30 BD B5 4A
28936 5A 26 F6 35 B6 34 36 10 8E 71 78 B6 01 39 26 05 8E 71 7D 20 03 8E 71 82
28960 C6 05 A6 A0 A7 80 5A 26 F9 35 B6 34 36 86 3F B7 FF 23 86 80 B7 FF 20 8D
28984 0C 7F FF 20 8D 07 86 37 B7 FF 23 35 B6 C6 64 5A 26 FD 39 AA AA AA AA 55
29008 AA AA 55 AA A6 55 9A A5 55 5A A6 41 9A AA 41 AA AA 41 AA 96 41 96 96 55
29032 96 95 55 56 96 55 96 96 55 96 AA 55 AA AA AA AA 00 00 00 00 00 00 00 00
29056 00 00 00 00 00 00 00 F5 5F DF D7 DF 77 DD F7 D7 F7 F5 5F F5 FF DD FF FD
29080 FF FD FF FD FF D5 5F F5 5F DF F7 FF F7 F5 5F DF FF D5 57 F5 5F DF F7 FF
29104 5F FF F7 DF F7 F5 5F FF 7F FD 7F F7 7F DF 7F D5 57 FF 7F D5 57 DF FF D5
29128 5F FF F7 DF F7 F5 5F F5 5F DF FF D5 5F DF F7 DF F7 F5 5F D5 57 FF F7 FF
29152 DF FF 7F FD FF FD FF F5 5F DF F7 F5 5F DF F7 DF F7 F5 5F F5 5F DF F7 DF
29176 F7 F5 57 FF F7 F5 5F FF FF
```

In search of atoms

Use your powers of deductive logic in ***Paul Hammond's*** *challenging game*

ATOM HUNT is an absorbing game to test your powers of deduction. The game uses the Dragon's high resolution colour grahics to the full to give an eye-catching game-board display. Text on the hi-res screen is produced by a sub-routine which draws "computer style" lettering. Most of all, the game requires intelligence and concentration, and becomes quite addictive.

When the program is run, an 8 × 8 grid is displayed. This grid (or molecule) contains four hidden atoms and your task is to locate these atoms. You have to deduce the locations of the four atoms by observing the deflections of light rays which you fire into the grid.

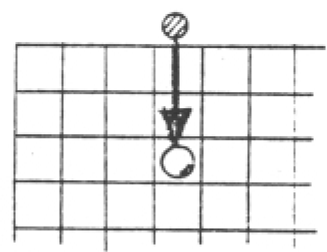


Figure 1: simple absorption

The computer does not reveal the path followed by a light ray — it only reveals the points at which the ray enters and leaves the grid. Each ray is represented by coloured entry and exit markers. In order to deduce the positions of the four atoms it is necessary to understand six laws of motion.

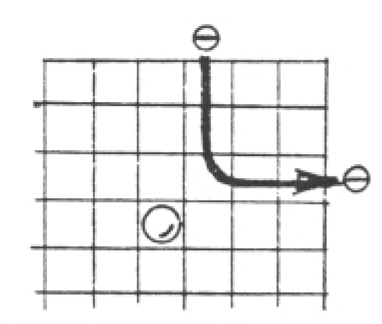


Figure 2: simple deflection

Simple absorption: Any ray which strikes a hidden atom "head-on" is absorbed and does not emerge from the grid. The computer indicates an absorbed ray by placing a cyan disk marker at the ray's entry point (see figure one). *Simple deflection:* A ray cannot pass alongside a hidden atom — it gets deflected at right angles as shown in figure two. In this case th computer places two identical markers to show the entry and exit points of the deflected ray. *Reflection:* When a ray approaches a pair of hidden atoms separated by one square, as shown in figure three, it is reflected back on itself and emerges from the grid at the same point that it entered. This reflected ray is indicated by a white marker disk.

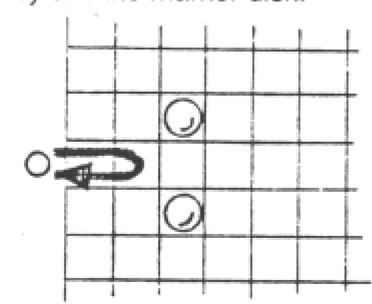


Figure 3: reflection

Reflection at an edge: If a light ray is fired into the grid at a position adjacent to a hidden atom, the ray is immediately reflected and so it is shown by a white marker disk (see figure four). *Absorbtion, not deflection:* If a light ray strikes one of a pair of adjacent hidden atoms, as shown in figure five, it is absorbed and the computer marks the ray with a single cyan disk. (Deflection from the adjacent atom does not occur.) *Clear path:* A ray travels in a straight line unless it is reflected, deflected or absorbed.

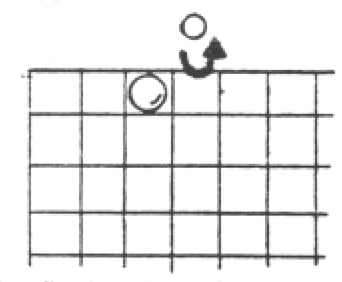


Figure 4: reflection at an edge

The illustrations show simple cases of deflection and reflection. In practice, a light ray may be deflected more than once on its journey. You must make allowance for this when guessing the locations of hidden atoms. Figure six illustrates various possible light paths.

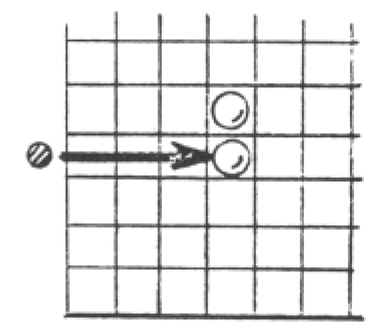


Figure 5: absorption not deflection

At the start of each new game the flashing cursor is positioned at the top left-hand corner of the grid. The cursor can be moved by using the arrow keys. To fire a light ray, first position the cursor anywhere on the border surrounding the grid and then press the "F" key. The computer will work out the path of the light ray and place markers as appropriate. The cursor should now be moved to a new position in the border and a second ray fired into the grid.

Before long, you will be able to deduce the location of one or more of the hidden atoms. As soon as this happens, you may mark the suspected square by "setting" a marker in it. You do this by driving the cursor to the suspect square and pressing the "S" key. Should you later change your mind, you can delete these set markers by driving to the suspect square once again and pressing "S" a second time.

When you are confident that your four set markers represent the actual locations of the four hidden atoms you should press the "G" key signifying "that's my guess". The computer will now reveal the real locations of the four atoms by painting four squares in cyan.

For every atom you guess correctly, you are awarded 10 points. But you lose one point for every light ray marker you used. The computer calculates and displays your score. The "highest score so far" is also displayed. Score ratings are as follows:

0-14	Poor
15-23	Fair
24-29	Good
30-32	Excellent
33-40	Unbelievable

The game is written in PMODE 3 to get high resolution with full colour. Text and graphics are mixed on the hi-res screen using a general-purpose sub-routine (line 1170) which writes any message, in any colour, at any screen position. The program is equipped with a full set of alphanumeric characters of constant height but variable width which gives the displays a touch of class. Readers may wish to adopt this character set for their own hi-res programs. The procedure for calling the print routine is illustrated in lines 520 and 1130.

The full game display takes several seconds to draw and is contained in lines 130-210. To save having to repeat this procedure for each new game, a "clean" copy of the starting display is held in the concealed video RAM pages 5 to 8 (see line 220). Then, at the start of each new game (line 240), the game display is copied down to the visible area, pages 1 to 4.

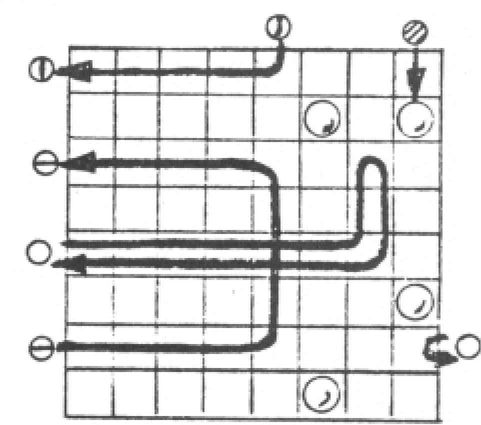


Figure 6: various possible light paths

The main inner loop is from line 260 to 390. The flashing cursor effect is obtained by a sequence which GETs a picture of the current square, PUTs in a cursor symbol ▶

◀ and PUTs back the original picture.

The keyboard is tested in line 270. Notice the use of the INSTR(L$,A$) function. L$ is defined at the beginning of the program (line 50) and is a list of legal keyboard commands. The INSTR function searches through this list until it finds the character that was typed and assigns a value to KEY. Then, in line 290, the program jumps to the required routine or returns to 260 if no key was pressed. This is a general-purpose structure which can be used in many programs.

The string arrays for the draw shapes are:

NN$(9)	Digits 0-9.
LL$(26)	Letters A-Z + space.
M$(16)	Light-ray markers.

The GET/PUT arrays are:

SQ(25)	Plain orange square.
CU(25)	Cursor (orange ring).
ST(25)	Store, for cursor flash.
AT(25)	Atom set by player.
AO(25)	Missed atom.
A1(25)	Correctly guessed atom.

And the other arrays are:

M(9,9)	Molecule array 1 = atom present.
G(8,8)	Guess array 1 = atom set by player.
T(3)	Turn factors for deflections.

Variables

NA	Number of atoms (4)
NG	Number of guessed atoms
SL	Side Length of grid (8)
SC	Score
HS	Highest Score
A9$,T9$,O9$,	DRAW strings for Atom
M9$	Hunt title block
W$	Word string for print routine
PX,PY	Cursor Position (0-9)
DC	Deflected coefficient
EF	Exit Flat
AF	Absorbtion Flag
VX,VY	Velocity of light-ray
LC	Light-ray counter
TE,UN	Tens and Units for score routine

Program notes

10-50	Initialisation.
60-70	Read in DRAW strings for numbers and letters.
80-90	GET shapes for blank squares, atoms and cursor.
130-140	Draw scoreboard panel.
150-180	Draw whole gameboard.
190	Draw notice board.
200	Draw "Atom Hunt" title block.
210	Put in the 8 × 8 grid . . .
220	. . . and display everything.
230	Start new game — clear arrays and set up molecule.
240-250	New gameboard; initialise variables.
260-280	Flash the cursor.
290	Jump on his command.
300-370	Cursor up, down, left, right.
380	Set/reset an atom.
390	Fire a light ray.
410-520	End of game + score routine.
550-590	Clear arrays for new game.
600-650	Set up atoms in a new molecule.
660-710	Set/reset atom.
720-960	Fire a light ray through the molecule.
970-990	Light ray turns left.
1000-1020	Light ray turns right.
1030-1040	Light ray is absorbed.
1050-1100	Light ray emerges, place markers.
1110-1160	Error message routine.
1170-1200	Hi-res print routine.
1210-1230	Hi-res digit routine.
1240-1290	Set up light ray markers in array M$(X).
1300-1330	Draw strings for digits 0-9.
1340-1390	Draw strings for letters A-Z.
1400-1460	Draw strings for Atom Hunt title block.

```
10 'ATOM HUNT BY PAUL HAMMOND.
20 'ISSUE 13 , JAN 1984.
30 CLEAR 2000:PCLEAR8:CLS4:PRINT @
 233,"PLEASE WAIT.";
40 DIM NN$(9),LL$(26),M$(14),SQ(25
),AT(25),AO(25),A1(25),A2(25),CU(2
5),ST(25)
50 T(1)=1:T(2)=4:T(3)=2:SL=8:NA=4:
L$="*"+CHR$(94)+CHR$(10)+CHR$(8)+C
HR$(9)+"SFG"
60 FOR J=0 TO 9:READ NN$(J):NEXT J
70 FOR J=0 TO 26:READ LL$(J):NEXT
J:READ A9$,T9$,O9$,M9$,H9$,U9$,N9$
80 A$="R2ND3R6ND3R2D1NL10D10L2NU3L
6NU3L2U1NR10U10":PMODE3,1:PCLS:LIN
E(0,0)-(11,11),PSET,BF:GET(0,0)-(1
1,11),SQ,G:PCLS:DRAW"BM0,0;C8"+A$:
GET(0,0)-(11,11),AT,G:PCLS:DRAW"BM
 0,0;C6"+A$:GET(0,0)-(11,11),A1,G
90 COLOR 6,5:LINE(0,0)-(11,11),PSE
T,BF:GET(0,0)-(11,11),AO,G:COLOR 8
,5:PCLS:LINE(0,0)-(11,11),PSET,B:L
INE(2,1)-(9,10),PSET,B:GET(0,0)-(1
1,11),CU,G
100 GOSUB 1250
110 DIM M(SL+1,SL+1),G(SL,SL)
120 PMODE 3,1:PCLS:CLS2:PRINT @ 23
3,"NOT LONG NOW";
130 FOR J=60 TO 136 STEP 38:LINE(2
8,J)-(45,J+11),PSET,BF:NEXT J:DRAW
"BM32,62;C5"+LL$(6):DRAW"BM32,100"
+LL$(19):DRAW"BM 32,138"+LL$(7):CO
LOR 8,5:XX=16:YY=78:W$="FIRE":GOSU
B 1180:XX=20:YY=116:W$="SET":GOSUB
 1180
140 XX=8:YY=155:W$="GUESS":GOSUB 1
180
150 COLOR 6,5:LINE(2,0)-(70,41),PS
ET,B:LINE(4,2)-(68,39),PSET,B:LINE
(2,1)-(70,1),PSET:LINE(2,40)-(70,4
0),PSET
160 LINE(2,53)-(70,171),PSET,B:LIN
E(4,55)-(68,169),PSET,B:LINE(2,54)
-(70,54),PSET:LINE(2,170)-(70,170)
,PSET:LINE(6,92)-(66,92),PSET:LINE
(6,129)-(66,129),PSET
170 LINE(82,0)-(254,171),PSET,B:LI
NE(84,2)-(252,169),PSET,B:LINE(82,
1)-(254,1),PSET:LINE(82,170)-(254,
170),PSET
180 LINE(102,19)-(232,150),PSET,B
190 LINE(2,178)-(254,191),PSET,BF
200 DRAW"BM8,7;C8"+A9$+T9$+O9$+M9$
:DRAW"BM8,23"+H9$+U9$+N9$+T9$+"BD1
1BL2U1"
210 FOR J=1 TO 8:FOR K=1 TO 8:X=10
6+(K-1)*16:Y=23+(J-1)*16:PUT(X,Y)-
(X+11,Y+11),SQ,PSET:NEXT K,J
220 FOR J=1 TO 4:PCOPY J TO J+4:NE
XT J:CLS:SCREEN 1,1
230 GOSUB 560:GOSUB 610
240 FOR J=1 TO 4:PCOPY J+4 TO J:NE
XT J
250 X1=90:Y1=7:LC=0:SC=0:PX=0:PY=0
:NG=0
260 X1=90+PX*16:Y1=7+PY*16:GET(X1,
Y1)-(X1+11,Y1+11),ST,G:PUT(X1,Y1)-
(X1+11,Y1+11),CU,PSET
270 A$=INKEY$:KEY=INSTR(L$,A$)
280 FOR J=1 TO 20:NEXT:PUT(X1,Y1)-
(X1+11,Y1+11),ST,PSET
290 ON KEY GOTO 260,300,320,340,36
0,380,390,410
```

continued on page 27

```
300 PY=PY-1:IF PY<0 THEN PY=0
```

ELITE*CALC

Features include

* **255 MAXIMUM NO. OF ROWS**
* **255 MAXIMUM NO. OF COLUMNS**
* **VERY EASY TO USE**
* **INDIVIDUAL CELL FORMULAE**
* **COPY BLOCKS OF CELLS**
* **COMPATIBLE WITH ALL PRINTERS**
* **EASY 132 COLUMN PAGE WIDTH**
* **GRAPH FORMAT FOR BAR CHARTS**
* **COMPREHENSIVE MANUAL INCLUDED**
* **INTERNATIONAL USER GROUP**
* **SORT ROUTINE IN ASCENDING OR DESCENDING ORDER**

* Single character commands
* Help display
* Enter text or formulae to 255 characters long
* Repeat text entries
* Available memory always displayed
* Rapid entry modes for text and data
* Selectable automatic cursor movement
* Insert, Delete, Move entire rows or columns
* Replicate one cell to fill a row or column with selectable adjustment
* All machine language for blistering speed
* >24K data storage space available in 32K systems.
* Basic style formulae
* Arithmetic operators: −, −, x, /, , (,), =
* Relational operators: =, >, <, <=, >=, <>
* Logical operators: AND, OR, NOT
* Conditional formulae: IF. . . THEN . . . ELSE
* Trig. functions: SIN, COS, TAN, ATN
* Log. Functions: LOG, EXP, SQR
* Misc. functions: INT, FX, ABS, SGN
* Range functions: SUM, AVERAGE, COUNT, MIN, MAX, LOOKUP
* Nine digit precision
* User definable constant table
* User definable printer set-up commands
* Individual column width settings (1 to 255)
* Adjustable row height to insert blank lines without wasting memory
* Hide columns or rows
* Alternate print font selectable cell by cell
* Display/Print formats set by cell, row or column
* £-format, comma grouping, prefix or postfix signs
* Scientific notation, fixed point and integer formats
* Left or Right cell contents justification
* Full page formatting
* All formats stored with worksheet on disc/tape
* Save/load disc/tape files in compact memory form
* Scan disc directories
* Output ASCII file for word processor input compatability
* Memory resident code . . . no repeated disc calls

*NOW . . . The worksheet calculator program you've been waiting for is waiting to work for you. ELITE*CALC is a powerful, full featured spreadsheet for the Dragon 32/64 and Tandy Color Computer. Answer 'what if . . .' questions, prepare reports and cash flow projections, maintain records and perform other tasks which, until now, required sophisticated business computers. ELITE* CALC is a serious tool for those who want to do more than play games on their micro.*

Availible from stock on cartridge for the Dragon 32 or 64 or Tandy Color Computer . . . please specify. Shortly available on disc for the Dragon Data or Cumana/Premier or Radio Shack systems. Also available in the $–format instead of the £–format if requested.

THE BEST FOR ONLY £45

Elite*Calc has had excellent reviews in the American press and an enthusiastic reception at the recent International Rainbowfest.

*"Elite*Calc is a great spreadsheet prograrn"*
Stuart Hawkinson, Rainbow

"Truly one of the best programs I have seen"
John Steiner, Micro

"Very powerful program . . . essential to every serious user"
Mike Jarvis, M & J Software

ENQUIRIES INVITED FROM RETAILERS AND DISTRIBUTORS (HOME AND OVERSEAS)

MICROCARE
1 OAKWOOD ROAD,
RODE HEATH,
STOKE-ON-TRENT
☎ (09363) 5695

DRAGON 32/64
TANDY COLOR COMPUTER

```
310 GOTO 260
320 PY=PY+1:IF PY>SL+1 THEN PY=SL+1
330 GOTO 260
340 PX=PX-1:IF PX<0 THEN PX=0
350 GOTO 260
360 PX=PX+1:IF PX>SL+1 THEN PX=SL+
1
370 GOTO 260
380 GOSUB 670:GOTO 260
390 GOSUB 730:GOTO 260
400 'HE'S MADE A GUESS
410 IF NG>NA THEN GOSUB 1120:GOTO
260
420 FOR J=1 TO SL:FOR K=1 TO SL
430 IF M(K,J)=0 THEN 460
440 XX=106+(K-1)*16:YY=23+(J-1)*16
450 IF G(K,J)=1 THEN PUT(XX,YY)-(X
X+11,YY+11),A1,PSET:PLAY"O4T20B":S
C=SC+10:ELSE PUT(XX,YY)-(XX+11,YY+
11),A0,PSET:PLAY"O2T20C"
460 NEXT K,J
470 IF SC<0 THEN SC=0
480 LINE(7,56)-(66,167),PRESET,BF
490 DRAW"C8":XX=8:YY=65:W$="SCORE"
:GOSUB 1180:XX=26:YY=83:GOSUB 1220
500 IF HS>SC THEN SC=HS ELSE HS=SC
510 XX=20:YY=116:W$="TOP":GOSUB 11
80:XX=8:YY=128:W$="SCORE":GOSUB 11
80:XX=26:YY=146:GOSUB 1220
520 XX=10:YY=181:DRAW"C5":W$="HIT
ANY KEY TO START":GOSUB 1180
530 A$=INKEY$:IF A$="" THEN 530
540 GOTO 230
550 ' CLEAR ARRAYS
560 FOR J=1 TO SL:FOR K=1 TO SL
570 M(J,K)=0:G(J,K)=0
580 NEXT K,J   590 RETURN
600 'SET UP MOLECULE
610 FOR J=1 TO NA
620 R1=RND(SL):R2=RND(SL)
630 IF M(R1,R2)=1 THEN 620
640 M(R1,R2)=1
650 NEXT J:RETURN
660 'SET/RESET ATOM
670 IF PX=0 OR PX=SL+1 OR PY=0 OR
PY=SL+1 THEN 710
680 IFG(PX,PY)=0 THEN G(PX,PY)=1:N
G=NG+1 ELSE G(PX,PY)=0:NG=NG-1
690 IFG(PX,PY)=1 THEN PUT(X1,Y1)-(
X1+11,Y1+11),AT,PSET
700 IFG(PX,PY)=0 THEN PUT(X1,Y1)-(
X1+11,Y1+11),SQ,PSET
710 RETURN
720 'FIRE LIGHT RAY  730 TIMER=0
740 X=PX:Y=PY
750 IFX>0 AND X<SL+1 AND Y>0 AND Y
<SL+1 THEN GOTO 1130
760 IF(X=0 AND Y=0) OR (X=0 AND Y=
SL+1) OR (X=SL+1 AND Y=0) OR (X=SL
+1 AND Y=SL+1) THEN 1130
770 EF=1:AF=0
780 IFX=0 THEN VX=1:VY=0
790 IFX=SL+1 THEN VX=-1:VY=0
800 IFY=0 THEN VX=0:VY=1
810 IF Y=SL+1 THEN VX=0:VY=-1
820 DC=0        830 FORJ=1TO3
840 IFM(X+VX-(J-2)*VY,Y+VY+(J-2)*V
X)=1 THEN DC=DC+T(J)
850 NEXTJ
860 IF DC=0 THEN 910
870 IFDC>3 THEN AF=1:GOTO910
880 IF DC=3 OR(DC=2 AND EF=1) OR (
DC=1 AND EF=1)THEN X=X+VX:Y=Y+VY:V
X=-VX:VY=-VY:GOTO910
890 IF DC=2 AND EF=0 THEN GOSUB 98
0:GOTO 910
900 IF DC=1 AND EF=0 THEN GOSUB 10
10
910 X=X+VX:Y=Y+VY:EF=0
920 IF AF=0 AND X>0 AND X<SL+1 AND
 Y>0 AND Y<SL+1 THEN 820
930 TI=TIMER:IF TI<40 THEN 930
940 IF AF=1 THEN GOSUB 1040:GOTO 9
60
950 GOSUB 1060
960 RETURN
970 'LEFT TURN
980 IF VX=0 THEN VX=VY:VY=0 ELSE V
Y=-VX:VX=0
990 RETURN
1000 'RIGHT TURN
1010 IF VX=0 THEN VX=-VY:VY=0 ELSE
 VY=VX:VX=0
1020 RETURN
1030 'ABSORBED
1040 DRAW"BM"+STR$(X1)+","+STR$(Y1
)+AB$:SC=SC-1:RETURN
1050 'RAY EMERGES
1060 X2=90+X*16:Y2=7+Y*16
1070 IF X2=X1 AND Y2=Y1 THEN DRAW"
BM"+STR$(X1)+","+STR$(Y1)+RE$:SC=S
C-1:GOTO 1100
1080 LC=LC+1:IF LC>14 THEN LC=1
1090 DRAW"BM"+STR$(X1)+","+STR$(Y1
)+M$(LC):DRAW"BM"+STR$(X2)+","+STR
$(Y2)+M$(LC):SC=SC-2
1100 RETURN
1110 'ERROR MESSAGES
1120 DRAW"C5":XX=44:YY=181:W$="TOO
 MANY ATOMS":GOTO 1140
1130 DRAW"C5":XX=10:YY=181:W$="CAN
T FIRE FROM THERE"
1140 GOSUB 1180:SOUND200,4:SOUND50
,4
1150 COLOR 6,5:LINE(2,178)-(254,19
1),PSET,BF
1160 RETURN
1170 'DRAW TEXT IN W$ @ XX,YY
1180 DRAW"BM "+STR$(XX)+","+STR$(Y
Y):FOR J=1 TO LEN(W$):ZZ=ASC(MID$(
W$,J,1))-64:IF ZZ<0 THENZZ=0
1190 DRAW LL$(ZZ):NEXT J
1200 RETURN
```

continued on page 28

```
1210 'DRAW 2-DIGIT SCORE @XX,YY
1220 TE=INT(SC/10):UN=SC-10*TE:DRA
W"BM"+STR$(XX)+","+STR$(YY)+";"+NN
```

```
$(TE)+NN$(UN)
1230 RETURN
1240 'SET UP MARKERS
1250 A$(1)="BD2D7BD2BR2NR6U1R6BR2B
U1U7BU1BL2NL6U1L6BL2":A$(2)=A$(1)+
"BD2BR2D7R2U7R2D7R2U7BL8BU2":A$(3)
="BD4BR2R6D1L6D1R6D1L6BL2BU7":A$(4
)="BR4BD2D7R2U7BL6BU2":A$(5)="BR4B
D4D3R2U3BL6BU4"
1260 RE$="C6"+A$(1):AB$="C6"+A$(2)
1270 FOR C=7 TO 8:M$(C-6)="C"+STR$
(C)+A$(2):M$(C-2)="C"+STR$(C)+A$(2
)+"C5"+A$(3):M$(C+2)="C"+STR$(C)+A
$(2)+"C5"+A$(4):M$(C+6)="C"+STR$(C
)+A$(2)+"C5"+A$(5):NEXTC
1280 FOR C=6 TO 8 STEP 2:M$(C/2)="
C"+STR$(C)+A$(1)+A$(3):M$(C/2+4)="
C"+STR$(C)+A$(1)+A$(4):M$(C/2+8)="
C"+STR$(C)+A$(1)+A$(5):NEXT C
1290 RETURN
1300 'DRAW STRINGS FOR NUMBERS
1310 DATA D7R2NU1R4U7L2ND1L2BR8,R2
D7L2R4BU7BR4,R6D3L6D4R2NU3R4BU7BR4
,R6D7L6R4U4NL2BU3BR6
1320 DATA D6R6L2D1U2BU5BR6,NR6D3R6
D4L2NU3L4U2BU6BR10,D7R2NU2R4U4L4BU
3BR8
1330 DATA R4D7R2U7BR4,D7R6U4NL4U3N
L4BR4,D3R4D4L4NU1R6U7NL4BR4
1340 'DRAW STRINGS FOR LETTERS
1350 DATA BR12
1360 DATA D7R2U3R6ND3U4D1L2U1NL6BR
6,D7R2U4R6D4L8U7R6ND3BR6,D7R2NU4R6
U1BU5U1NL8BR4,D7R2NU4R6U6D1L2U2NL6
BR6,D7R2NU4R6U1BU3BL2L6U3R8ND1BR4,
D7R2U4NR4BU3R6ND1BR4,D7R2NU4R6U4NL
2BU2U1NL8BR4
1370 DATA D7R2U4R6ND4U3BR4,D7R2U4B
U3BR4,BD5D2R2NU2R4U7BR4,D7R2U4R4NU
3R2ND4BU3BR4,D7R2NU4R6U1BU6BR4
1380 DATA D7R2U3BU4R4ND7R6ND7BR4,D
7R2U4BU3R6D1NL2D6BU7BR4,D7R2NU4R6U
7D1L2U1NL6BR6,D7R2U4R6U3NL8BR4,D7R
2NU4R8U1L2U6NL8BR6,D7R2U4R6ND4L2U3
NL6BR6,D3R8BU2NL2U1NL8BD3D4L6NU2L2
U2BU5BR12,R2D7R2U4BU3R4BR4
1390 DATA D7R2NU4R6U7D1L2U1BR6,D5R
2NU3ND1R2ND2R2ND1R2U5BR4,D7R2NU4R4
NU7R6U7BR4,D2BD2D3R2U4NU3R4NU3ND4B
R2BD1ND3BU2U2BR4,D3R2D4R2U2BU2R4U3
BR4,ND1R2ND1R6D1BD1BL1L1D1BL1L1D1B
L1L1D1BL2D2R2NU3R4NU1R2U1BU6BR4
1400 DATA D11R2U5R8D1NL8D4U11D2L2U
2L6D1R6U1BR6
1410 DATA R10D1L10R4D10R2U7BR8BU4
1420 DATA D11R2NU7R8U1NL6U10D2L2U2
L6D1R6BU1BR6
1430 DATA D11R2U6BU5R4D1NL4ND10U1N
R6D1R4ND1R2D10BU11BR4
1440 DATA D11R2U6NR8D1R8D5U11BR4
1450 DATA D11R2NU8NR8U1R8U10BR4
1460 DATA D11R2U7BU3R6ND1R2L8U1R8D
11BR4BU11
1470 END
```

Machine code routine for recovering files

Using 14 bytes of machine code **Pam D'Arcy** *explains how to recover a file before the dreaded 10/ERROR strikes*

A READER threw down the gauntlet earlier this year, asking if it was possible to recover a file, when Tapescan (February, *Dragon User*) shows that many blocks are perfectly allright before the dreaded IO ERROR strikes. Well, the answer is yes.

The key to this is so simple as to be almost unbelievable — 14 bytes of machine code that have appeared in many places including *Dragon User* (July's issue, under the deceptive title of "Loading Hex" for such an invaluable item).

The Basic program is read from cassette into memory starting at the memory address in location 25 (hex &H19). (The value of this location depends on the last PCLEAR issued/POKE 25,6 business.)

Instructions

August's issue described how Basic instructions are held in memory. You may recall that the first two bytes of an instruction contain the memory address of the start of the next instruction. Each instruction is terminated with a null (&H00) byte. The end of a program is indicated by the two bytes following the last program instruction (that is the location pointed to by the address pointer field at the start of the last instruction) being set to nulls. During editing of program instructions, and execution of PCLEAR, which may change the start position of the Basic program, the linking address pointers within the instructions change. Recovery of Basic text is achieved by taking advantage of the Basic ROM code that resets these program address pointers.

The code given below is often referred to as a method of recovering a Basic program in memory if you have inadvertently entered NEW, only to perhaps then realise that the CSAVE had been ineffective (either the speed up POKE was in effect or perhaps, as I have often been known to do, the record button was not set on the recorder — you name it, it can happen).

Typing in NEW sets the first two bytes of the program area (addressed by &H19) to nulls. Providing that no new Basic instructions have been entered (nor, I assume, PCLEAR has been changed when those bytes were null as it would think that there was no text to be repositioned) the magical 14 bytes of machine code resets the program address pointers, including the first two bytes, and . . . the LIST reveals that the program has re-appeared. However — it is *also* the answer to reclaiming a Basic program as far as the IO ERROR!

Inspection of the text area after such an error shows that the first two bytes have been set to nulls, resulting in OK if you type in LIST. Obey the 14 bytes, and a program listing will appear! The end of the program may have a "junky" look to it (although I have had no problems) as, of course, the Basic interpreter is relying on finding a null byte to terminate an instruction and there needs to be three null bytes together to determine the end of the program, so it will be interpreting anything that is sitting in RAM beyond the genuine program text (and may even display some "out of sequence" line numbers that you won't be able to access). Deleting text from the last genuine line number to end (DELnnnn–) will clean up the end of the program text.

The magical 14 bytes may be entered using Peter Barry's original contribution (July), Assemblers, Monitors, Topsy (June), Bruce Devlin's December item or direct POKEs — my word, such choice.

If the program is not in memory at the critical time and it cannot be CLOADM'd for any reason, POKE it in byte by byte to a "safe" part of memory (around 32000 will probably be fine) and EXEC to its start address. Mind you, as it is a CLOAD error that we are getting over, it won't matter about repeating the entire process from scratch anyway, having preloaded the recovery code which could then be the original Basic loader if necessary.

Reserving

CSAVEM the program. Before loading (and in fact setting it up depending on chosen method) do not forget to CLEAR to reserve the machine code area. To run it, EXEC (with its start address if you have been using other machine code routines as you may have executed a different routine by mistake!).

As the Basic program is stored without synchronisation gaps between blocks on the cassette, recovery beyond the IO ERROR will require more ingenuity. However, even to have rescued the first part of a program may save a lot of heartache!

To recover machine code programs/machine code data files, the program/data is read from cassette to memory starting ▶

```
                 * Recover Basic program after NEW
                 * Rescue Basic program after I/O ERR
                 *
                 * Fully relocatable - place code where it is best for you
7D4F              ORG  32079  will then directly precede Tapescan
7D4F 9E 19        LDX  $19    address of start of Basic text
7D51 BD 83 F3     JSR  $83F3  reset address pointers in Basic text
                 * Reset Basic workspace address pointers
7D54 30 02        LEAX 2,X    address foll. 2 null bytes @ end of program text
7D56 9F 1B        STX  $1B    beginning of simple variables storage
7D58 9F 1D        STX  $1D    beginning of Array Pointer Table storage
7D5A 9F 1F        STX  $1F    end of storage in use (= first free byte)
7D5C 39           RTS         Exit from recovery program
```

The recovery program listing

◀ at the address shown by the original CSAVEM, plus OFFSET if given. This, as with the Basic program, is as far as the IO ERROR is safely input. It will need some sort of inspection facility, for example, Topsy (June), monitors, dissassemblers, to determine how much program/data is recoverable. Once determined, it can be safely CSAVEM'd and reloaded at any time for reconstitution of the lost code/data (using Topsy/monitors/specially written or adapted programs and so on).

Data files

Recovering non-machine code data files, naturally, depends on the data. If the error occurs in a data file on a purchased tape, the best answer is to try and obtain a replacement as it may otherwise make nonsense of a game, for example.

Your own created files are an easier proposition. Whether you go about it by writing a special program or otherwise is up to you, but you may be interested in a technique that I have devised for programs that update files containing any "shape" of string data. I will keep the description as concise as possible, so I hope that you can follow the gist all right.

File format: Data files that are written to tape commence with three numeric fields:

(I) Number of data items being saved (records*fields R*F).
(R) Number of data records being saved.
(F) Number of fields per record.

Data array: The data is set up in/saved a single DIMensioned array (for example DIM RECS$(500)). When creating/loading a file, field titles are stored in subscripts 0–(F–1) and data records commence at subscript F.

The highest record number currently held in the array is set up/maintained during processing (R). Any access to the start of a record can be made by subscript N where N=(required record num*F), and any field within the record can be accessed by adding the required field number–1 to the subscript. Files set up like this are compatible with Dragon Data's Database program (Special Selection 2) enabling its facilities to be used on any string data files.

Saving the data: The fields I,R,F are written. The array is then written, commencing with subscript 0 (if no field titles are required, dummy null strings are written for compatibility), using a FOR ... NEXT loop until subscript ((R+1)*F)–1) has been written. Automatic rescue of valid data when reading in an old data file and an IO ERROR occurs: Clear an indicator (say, FLAG=0); open the file and input the numeric fields into variables, say, I,R,F. Set FLAG to, say, 1 to indicate that the program is in the "read data" situation. Read the data into the array, such as:

```
FOR V=0 to (I–1)
INPUT#–1,RECS$(V)
NEXT V
CLOSE #–1:FLAG=0:REM file read OK.
```

IO ERROR occurred: Enter GOTO nnnn where nnnn is the automatic recovery routine. Whatever you do, don't use RUN or the data will be lost!

For the automatic recovery routine:
CLOSE #–1
If FLAG=0 then there is no further recovery required/possible (didn't occur during reading of data fields). Go to main menu process or whatever to continue.
If FLAG=1, recover to last complete data record input.
V=the subscript of the data item being read at the time of the failure, therefore R(current highest rec.in the array)=INT(V/F)–1 (If R=0 after this calculation, an error occurred before a single complete data record was fully read in).

Go to the main menu process or whatever (or branch automatically to your "save data file" facility to secure the recovered data and to prevent another disaster from striking)!

Compatibility

This style of data handling copes with any "shape" of data file (differing numbers of fields/record) keeps it compatible enough to use the facilities of readily available Database software and allows you a standard piece of recovery code. Furthermore, my typed in GOTO after the IO ERROR is always to line 4 (do you want to know what I use 1,2 and 3 for)? Line 4 contains the relevant GOTO for this program so that I don't need to worry about having renumbered/what version/program I actually have in memory at the time — the stress of an IO ERROR can be sufficient without accidentally GOing TO the wrong place! ■

Speaking in dialects

***Keith and Stephen Brain** review OS9 languages*

IN ADDITION to the compiled Basic09 language, reviewed earlier in *Dragon User*, Dragon Data is also marketing two other language packages running under OS9 — Pascal V2.0 and C-Compiler (£79.95 each). These offerings are rather unusual for such a small system but they extend the capabilities of the Dragon far beyond its humble origins and towards the far loftier heights of much more powerful hardware and software concepts.

Pascal

Pascal is a language much loved by computer scientists because of its inherently logical structure, great power, and general "correctness" in their eyes. It was originally developed in the late 1960s by Professor Niklaus Wirth of Zurich, as a means of teaching programming as a logical and systematic discipline, and like all good languages now has a series of dialects. The dialect used here follows the ISO specification, rather than the UCSD model. Some of the more obvious original distinctions between Pascal and Basic have become rather blurred in some of the more recent and powerful versions of Basic (especially Basic09) as many of the best features of Pascal have been transported across.

A major difference to the programmer used to standard Microsoft Basic is that Pascal programs must be completely written with some form of text editor, and then compiled into an intermediate form, known as P-code, by a Pascal compiler before they can be run and tested. Whilst a comprehensive debugging package helps sort out the bugs which inevitably tend to fall into your programs, we find working with this type of batch-compiled language very tedious. I know that the "experts" will (perhaps quite rightly) say that this is because we are sloppy programmers, but in our experience it is often the empirical approach which seems to work best.

Pascal has found very wide application in serious computing because of its versatility and power, but, power almost inevitably implies size and there have therefore been difficulties in the past in trying to fit full-feature versions of Pascal into the pint pot of a microcomputer. However, as the 6809 microprocessor was specifically designed from square one to run such high level languages, life with the Dragon is more tolerable than usual. Programs compiled into P-code run rather slower than pure machine code, as each instruction must be processed in turn by the run-time interpreter, but, the OS9 Pascal goes further than usual and allows you to also convert this P-code directly into "native" 6809 machine code. Taking this "back-door assembler" route gives a speed advantage of some four to 10 times over standard Pascal, without the necessity of ever actually grappling with mnemonics!

Another major factor in the attraction of this particular package is the ability of the OS9 operating system to support "virtual memory" on disk. This means that you can actually run Pascal programs which are much bigger than the total memory size. A good example of the use of this feature is the Pascal Compiler itself, which operates in this way by swapping blocks into and out of memory. The language is supplied on two disks, both of which are needed to run the language, so a double disk drive is essential. "Pascal" is a machine code front-end which calls "PascalS" (the swapping P-code interpreter) to run the P-code "Pascal-Compiler".

The text file "PascalErrs" produces full English error messages at all stages. "PascalN" is used to run compiled P-code programs, unless they are so large that the swapping method (and "PascalS") must be used, which adds a time penalty. "PascalT.RUN" is the native code translation program, which is written mainly in P-code but also calls some machine code routines from "PascalT.MODL", and uses the "PascalDefs" file which contains assembly language source code definitions. "PascalE" is a linkage editor which is used to combine separately compiled procedures into a single program. Three machine code support modules containing commonly used library routines are also included. "Support" takes up 9K, but two alternative stripped-down versions are also provided ("Support1" (7K) and "Support2" (5K)).

An extensive User Manual is included in the price, but if you are a newcomer to Pascal then you will also need a good introductory book — of which there are many (although *Programming in Pascal, Revised Edition* by P Grogono (Addison-Wesley, 1980) and *Introduction to Pascal — second edition* by J Welsh and J Elder (Prentice-Hall, 1982) can be recommended). If you want to learn Pascal, or the language has obvious advantages in your particular applications, then this comprehensive package does all you could reasonably ask of it. For ourselves we will probably continue to be peasants who prefer to use Basic09, which has many Pascal-type features but a more user-friendly compiler interface.

C-Compiler is a rather more recent development than Pascal, having emerged from Bell Laboratories in 1972 as Dennis Ritchie's development of an earlier language named "B" (who said that computer scientists had no imagination?). The main feature which makes "C" stand out from other languages is that it was designed from the outset as a means of writing "portable" programs. In this context portability refers to the ability to run a program on different machines rather than any question of physical size. It falls somewhere between high-level languages like Basic and Pascal and Assembly Language, providing a workable structure which is close to machine code but essentially processor-independent. The fundamental flow-control constructions (if, while, for, do and switch) are supported, but "C" deals essentially with characters, numbers and addresses. Inevitably it is not the easiest language to learn, and it does not feature all the error traps of higher-level languages, but you can't make omelettes without breaking eggs. The "Bible" of the "C" programmer is *The C Programming Language* by B Kernigham and D Ritchie (Prentice-Hall, 1978), although the price of £17.95 is rather steep and *Learning to Program in C* by Thomas Plum (Prentice-Hall, 1983) is both cheaper (£12.95) and more readable.

Growth

A major demonstration of the power of the language is the fact that the Bell Unix operating system (on which OS9 itself is based) was entirely rewritten in "C" by Ritchie so that it could be routinely run on IBM, Honeywell and Interdata systems. "C" is rapidly growing in popularity amongst serious software writers as it makes them more productive. Once a "C" program is written it can easily be "ported" on to any machine which has a "C-Compiler" available, and in particular it is claimed that because of the close similarity between OS9 and Unix almost any application written in "C" can be directly transported, recompiled and correctly executed.

The OS9 C-Compiler again comes on two disks, together with a comprehensive manual. There is no official standard for "C" but this version follows the Kernighan and Ritchie model closely (but with some enhancements and extensions). In particular the ability of the 6809 to use a "direct page" structure is supported, and assembly language may be embedded. The system interface supports almost all the system calls of both OS9 and Unix and a complete standard library of predefined standard functions is included ("stdio.h").

The "ccl" command calls a two pass compiler ("c.pass1" and "c.pass2") which converts source code into an executable file. An optimisation ("c.opt") pass automatically occurs after the compilation passes, which removes redundant code and searches for sequences that can be replaced by shorter and faster equivalents. A profiler option can be included which ▶


What's your best source of information on color computing?
HOT CoCo
Now you can improve your color computing skills... and it's easy to do. HOT CoCo gives you more practical information on the Dragon* than any other publication. Nearly 150 pages a month!
Every issue is packed with exciting new things for you to do. We won't waste your time with filler stories. You'll get instructive columns:
•Elmer's Arcade—enjoy old-fashioned arcade style games on your computer
•The Basic Beat—learn everything you need to program in Basic
•The Educated Guest—discover how to use your computer as a teaching tool
•Doctor ASCII—get answers to your technical questions
•Graphically Speaking—create eye-catching designs that add appeal to your programs
You also get a dozen easy-to-understand articles every month. Games... utilities... programming techniques... tutorials... graphics... education... hardware projects. They'll help you expand what you can do. And complete program listings show you how to use what you learn.
That's not all. HOT CoCo saves you money too:
•Candid reviews help you make every purchase a sound investment.
•Informative ads let you comparison-shop from home.
•New-product announcements tell you what's available before it reaches the stores.
With all this at your fingertips, your subscription could pay for itself with one wise purchase.
And HOT CoCo is risk-free. If you don't like your first issue, just write "cancel" across the invoice and return it to us. You won't owe a thing.
Subscribe to HOT CoCo today. Twelve big issues are only $44.97 (US funds drawn on a US bank). Simply fill out the coupon below and return it right now to: HOT CoCo Subscription Dept., PO Box 975, Farmingdale, NY 11737, USA.
HOT CoCo
Make Your Children Smarter
* Dragon is a registered trademark of Dragon Data Ltd.
YES! Help me improve my computing skills. Send me 12 issues of HOT CoCo for $44.97 (US). I understand that with payment enclosed or credit card order I will receive a FREE issue, making a total of 13 issues for $44.97 (US).
Get a 13th issue FREE when you enclose payment or charge it on your Mastercard, Visa, or American Express.
☐ CHECK/MO ☐ MC ☐ VISA ☐ AE
CARD #______ EXP. DATE______
SIGNATURE______
NAME______
ADDRESS______
POST CODE______ COUNTRY______
HOT CoCo • 80 Pine Street • Peterborough, NH 03458 • USA
73DFDU

◀ counts each time a function is called during execution, so that program structure can be logically modified if desired. The final output is position-independent reentrant 6809 code in standard OS9 memory module format. This code can be used as a subroutine called from the Basic09 RUN command, although care must be taken as internal data representation is not identical. If you are one of the stout hearts who can see into the future and wants to get to grips with "C", then here is your chance to get into the act at a bargain system price, and write software compatible with the next generation of machines.

The final utility disk currently available from Dragon Data is the Editor Assembler Debugger package which comes on a single disk, with a comprehensive manual, for £49.95. The first part is a powerful macro text editor. Although the manual suggests that "it is commonly used to prepare letters and documents" we feel that you must be rather a masochist to use it in preference to a proper word processor such as Stylograph! On the other hand it is extremely useful for preparing program source files for Pascal, "C" and the Assembler itself.

Operations

Multiple read/write files can be open simultaneously, all OS9 commands are usable within the workspace, and the editor commands are a superset of those used in Basic09. Search and replace operations are supported, conditional tests can be applied and edit macros can be defined as new commands to perform particular specialised tasks. The Assembler was designed specifically for the modular multi-tasking environment of OS9 and therefore incorporates features for calling OS9, generating memory modules, encouraging the creation of position-independent code, and maintaining separate program and data sections. A tree-structured symbol table organisation provides fast assembly speed and it has been optimised for use with the "Pascal" and "C" compilers.

In addition to producing "normal" OS9 modules the assembler can also produce "Motorola-compatible" code which is suitable for the standard Dragon, and conditional assembly is possible with IF, ELSE and ENDC. This disk (and the system disk) holds DEFS files containing labels with their associated values which can be used directly for system calls thus making life simpler and more logical whilst saving much brainwork or thumbing through the manual for codes. Error messages are printed out in the listing just below the source line containing the error. The Interactive Debugger (DEBUG) is the final part of this trio, providing calculations, memory examine and change, register display and change, breakpoint set and remove, memory clear and test, memory dump and memory search, and programs can be executed in a number of ways. Finally the Shell command allows system commands to be passed and other programs to be manipulated from within the debugger.

```
LOAD          Load module(s) from a file        F$LOAD

ASSEMBLER CALL    OS9 F$LOAD

MACHINE CODE      103F 01

INPUT:  (X) = Address of pathlist (file name)
        (A) = Language/type (0 =any language/type)

OUTPUT: (X) = Advanced past pathlist
        (Y) = Primary module entry point address
        (U) = Address of module header
        (A) = Language/type
        (B) = Attributes/revision level

ERROR OUTPUT:   (CC) = C Bit set
                (B)  = Appropriate error code

Opens a file specified by the pathlist, reads one
or more memory modules from the file into memory,
then closes the file.  All modules loaded are
added to the system module directory, and the
first module read is LINKed.  The parameters
returned are the same as the LINK call and apply
only to the first module loaded.

In order to be loaded, the file mus have the
'execute' permission and contain a module or
modules that have a proper module header.  The
file will be loaded from the working execution
directory unless a complete pathlist is given.

Possible errors: module directory full; memory
full; plus errors that occur on OPEN, READ, CLOSE
and LINK system calls.
```

An example of one of the Service Request Descriptions

The basic starting OS9 package consists of the OS9 System Disk and a detailed *OS9 Operating System User's Guide* for £39.95. A further even weightier tome, the *OS9 Operating System — System Programmer's Manual,* is also available from Dragon Data, but only in exchange for a further 50 per cent on the purchase price (£19.95), although that does include yet another of those video cassette cases! So what is the essential difference between a "user" and a "system programmer" and do you really need the information in the second volume? Perhaps the answer is already there, to some extent, as the very fact that the parts are sold separately indicates the non-essentiality of the further information to many users.

The *System Programmer's Manual* is of a "general" nature, describing implementation of OS9 on any hardware, a factor which can sometimes cause confusion as it goes into details on ROM contents. It does, however, set out clearly the details of Basic System Organisation, Kernel Functions, Memory Utilisation, Multiprogramming, Process Creation, Execution Scheduling, Signals, and Interrupt Processing, before going on to the structure and definition of memory modules.

The Unified Input/Output system is described in detail with explanations of the operations of the File Managers, Device Driver and Descriptor Modules, Random Block File Manager, Disk Organisation, File Descriptors, Device Descriptors and Drivers, Sequential Character File Manager, Line Editing and so on. These sections are mainly concerned with new implementations, although they are also essential reading if you want to add any "non-standard" devices to your Dragon. A brief mention of Assembly Language Programming Techniques is included, together with information on Adapting the Initialisation Module. A major (and probably the most important) part of the book is the lengthy series of Service Request Descriptions which define the service calls which are used to communicate between the OS9 operating system and assembly language programs. All these system calls have a mnemonic name beginning with "F$" for system functions or "I$" for input/output related requests, and they can be called by the "OS9" directive of the Assembler.

Undoubtedly this manual is essential if you are going to do any serious Assembly Language Programming, although its appeal to the average user is far more restricted.■

1. AVOID THUNDERSTORMS.

By linking into Prestel, you could call up detailed weather reports at any time of day or night.

2. CHECK THE SPELLING OF EVERY WORD YOU WRITE—INCLUDING THE TECHNICAL ONES.

If you're not too sure of your spelling, the Spellcheck program will put you right in seconds.

5. STOP WORLD WAR III BEFORE IT STARTS.

Naturally, there are literally hundreds of computer games to while away the extra spare time your GEC Dragon 64 has created for you.

6. FIND A CURE FOR INSOMNIA.

Instead of lying awake worrying about the business, get the GEC Dragon 64 to keep it all under control.

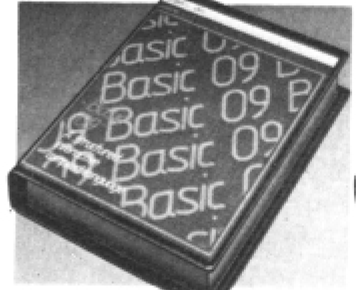
8. SEND REPORTS OVER THE PHONE.

You can send urgent messages or information through Prestel to the GEC Dragon 'Mailbox', for collection by other computer users.

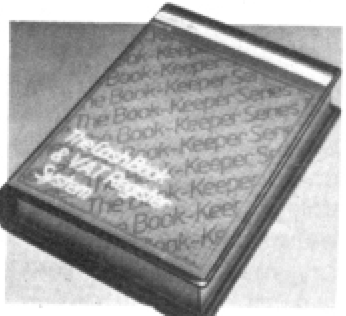
9. SPEND SUNDAY MORNING IN BED.

The biggest benefit of them all if you're in business on your own. By taking care of all the details, the GEC Dragon lets you concentrate on the more important things in life.

"What would I do with a GEC Dragon 64?"









3. WORK OUT HOW MANY TINS OF CAT-FOOD YOU HAVE IN STOCK.

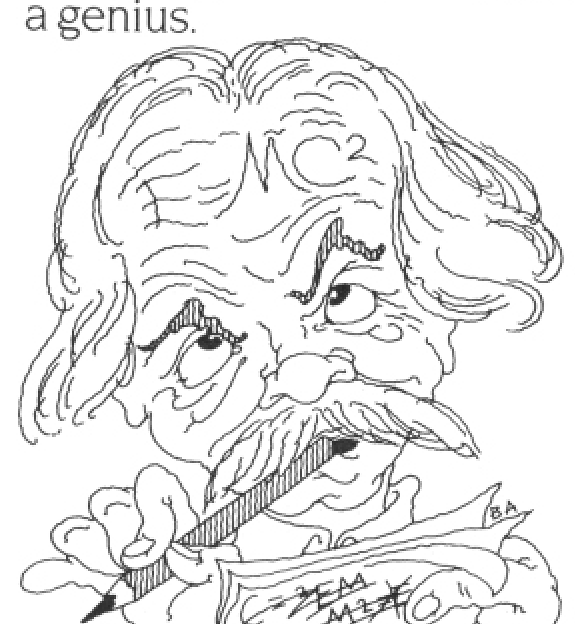
And work out which are the fastest and most profitable lines.

4. CONTACT EVERY ONE OF YOUR CUSTOMERS.

Many businesses use GEC Dragon's Mailmerge program to type the same letter, personalised to suit every one of thousands of customers. All you do is write the basic letter, give it the names and addresses, then sit back and wait for the replies.

7. CHECK THAT EINSTEIN GOT IT RIGHT.

When it comes to advanced maths and formulae, the GEC Dragon is little short of a genius.



10. WORK OUT WHAT YOU'LL BE WORTH WHEN YOU RETIRE.



Play the investment and insurance companies at their own game and work out EXACTLY how big your nest egg will be when the great day arrives.

11. SPEND AN EVENING WITH NEIL DIAMOND.

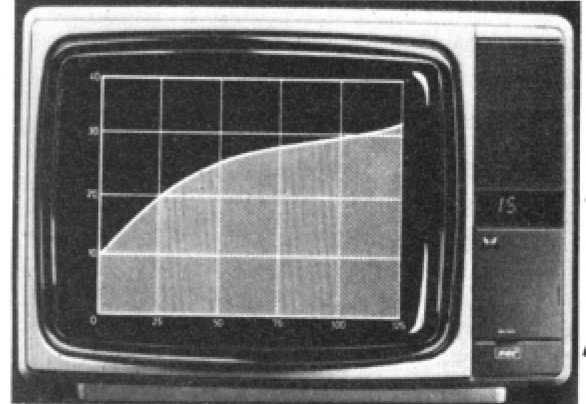
With a little help from Prestel, you can book seats at almost any show or theatre without even leaving your armchair.



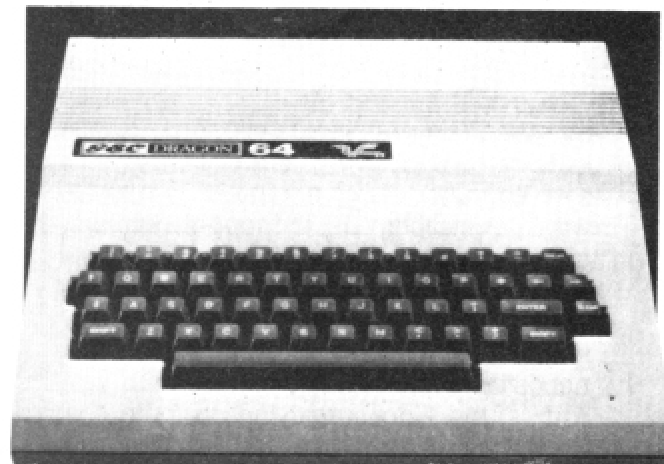


13. WRITE A THESIS.

If you're not very good at typing, or keep changing your mind, the GEC Dragon word processing program lets you edit, change, add extra pieces and delete. Then when your masterpiece is finally ready to type, just press a button and sit back.









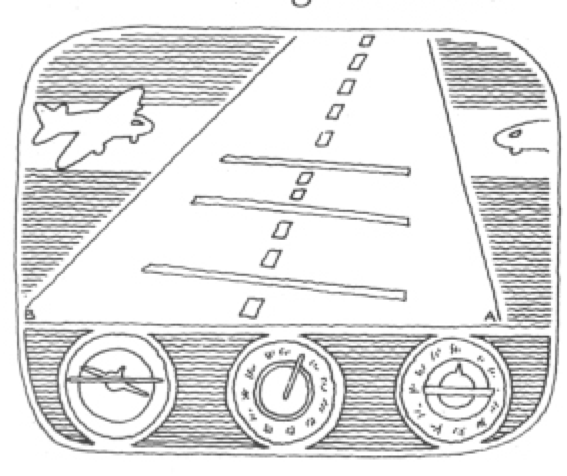
14. CLAIM YOUR FORTUNE ON THE POOLS.

The GEC Dragon 64 also gives you immediate access to a mass of sports information available through Prestel.



12. LEARN TO FLY A PLANE.

We even know someone who has created their own program to simulate the controls of a light aircraft.



15. BOOK YOUR HOLIDAYS.

Check the availability of practically any holiday you care to think of. Then make a reservation on the one you like best.



But that's just for starters. Later, we'll show you lots more ways the GEC Dragon 64 can make life simpler.

You can buy the GEC Dragon computer and a wide range of accessories and software from the better computer shops, major stores and GEC dealers.

It's proof that, now GEC and Dragon have got together, we're really going to start turning it on for the small business and serious computer user.

And to whet your appetite still further, we've produced a 12-page colour brochure that tells you how to get the most out of a GEC Dragon 64. It's called 'Your passport to professional software.'

It's yours free in exchange for the coupon below.

GEC DRAGON COMPUTERS

To: GEC Dragon Customer Services, Tripsgate House, Gladstone Drive, Staple Hill, Bristol BS16 4RU.
Please send me a copy of 'Your Passport to Professional Software.'

Name ______________________

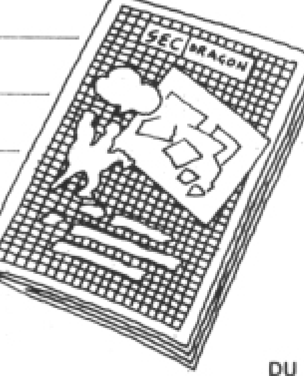
Address ______________________

____

Postcode ______________________

Or if you would like information on the rest of our products – please tick the appropriate box.

☐ Dragon 32 ☐ Dragon 64 ☐ Dragon Accessories



DU

Dragon library

***Mike Harrison** reviews a selection of books for the Dragon*

THE DRAGON 32 certainly comes out as one of the best middle-priced micros, but its manual has been attacked for being one of the worst. The reader is assumed to already understand the main principles of computing and therefore it is really only useful to those who don't really need it.

The combination of the popularity of the machine, the paucity of the documentation and the versatility of the excellent 6809 chip, has led to a veritable explosion of books being published to fill the gap. With this article, 36 titles are listed, covering areas as diverse as a computer songbook to lists of games, and from children's primers to assembly language programming. If you add to these the titles I've probably missed and the wealth of material published for the Tandy Colour Computer there is more than something for everyone. My thanks go to the Dragon Users' Club for help in compiling this list — and if you know of any other useful books please let *Dragon User* know.

Apart from the restricted choice on the shelves in large stores, the main access to these books is through mail-order. So how do you know what to expect in any book and are they all the same? I've taken a few off the list, read them thoroughly, typed in some of the listings and sifted out some of their gems which may be of interest to Dragon Users, both newcomers and old hands. I've tried to give some of the flavour of each book and to describe its contents so that you can judge for yourself if it might meet you needs. I'll begin with primers and then move on to look at the books for more advanced users — some will have to wait for future articles.

The introductory books assume no previous knowledge of Basic and help you into the world of computing through your Dragon. One such book, written specifically with children in mind, is Richard Wadman's *Dragon Magic*. It follows an effective path towards simple aims introducing children to the power of their machines yet re-inforcing their user friendliness.

Readers are advised to work through the book slowly and make sure that they understand each section before moving on to the next. Children should ask a parent or teacher if there is something they don't understand, says Richard, or write to Dragon Data whose address is given.

Each chapter ends with a list of things to remember, summarising the teaching points so far. The author emphasises the importance of sequencing of events in his section on program order, flowcharts and loops, which is necessary to establish at an early stage the importance of planning.

Many educationalists are highly critical of the almost universal adoption of Basic (meaning Beginner's All-purpose Symbolic Instruction Code) as the computer language that children start on. They claim that the main reason it is chosen — that it is close to English and easy to learn — is also its major disadvantage. It is easy to write unstructured, ill-thought-out programs which actually work in Basic but when they take students into university and industry this indiscipline hinders their learning to program in the rigorous ways necessary.

It is therefore gratifying to note that *Magic* is stressing the planning stage early in the lives of our perhaps future systems analysts.

Amusing little cartoon illustrations pop up on each page or so to emphasise some graphic point. "Keep wires tidy" and "do not poke around inside your computer or tv set" say the figures and later a number of sketches of shoe boxes are utilised to demonstrate the meaning of variables, one of the hardest concepts for young children to understand. String variables are hung on a washing line.

The book is very readable and its format of coaxing the reader a little at a time towards understanding by using ideas in the text, in illustrations, in small programs and finally as a "rule" mimics the best ways of learning.

Richard uses this method to take children to hi-res, demonstrating LINE and CIRCLE. He ends the book with a glossary of the commands he has covered and with 16 small demonstration programs. These programs (average 10 lines each) will only take 10 minutes each to type in, yet point

Read all about It

Advanced Sound and Graphics for the Dragon 32	Keith and Steven Brain	Sunshine	£5.95
Anatomy of the Dragon	Mike James	Wiley	£6.95
All About Forth	M Haydon	M & J Software	£7.95
Boots Guide to the Dragon 32	Ian Sinclair	Granada	£2.95
The Color Computer Songbook	R Clark	Arksoft	£6.45
The Dragon Companion	M Jarvis		£4.95
Dragon Extravaganza	Roger Valentine	V & H Comp Services	£4.95
Dragon Machine Code for the Absolute Beginner	John Vander Reyden	Melbourne House	£6.95
Dragon Magic	Richard Wadman	Foulsham	£4.95
The Dragon Programmer	S M Gee	Computer Bookshop	£5.95
The Dragon 32 Book of Games	James, Gee, Ewbank	Granada	£5.95
The Dragon 32 And How To Make The Most Of It	Ian Sinclair	Granada	£5.95
Dragon 32 Machine Code For Beginners	Mike James	Compudat	£5.95
Dragon 32 Games Master	Keith and Steven Brain	Sunshine	£5.95
Dragon 32 Programmers Reference Book	John Vander Reyden	Melbourne House	£6.95
The Dragon Trainer	Brian Lloyd	Sunshine	£5.95
Dynamic Games for the Dragon 32	Young, Bush and Shrimpton	Interface Publications	£4.95
Easy Programming for the Dragon 32	Stewart and James	Shiva	£5.95
Enter the Dragon	Carter	Melbourne House	£5.95
Further Programming for the Dragon 32	Stewart and James	Shiva	£5.95
Getting The Most From Your Dragon 32	David Bannister	Penguin	£4.95
Introducing Dragon Machine Code	Ian Sinclair	Granada	£7.95
Inside the Dragon	Smeed and Sommerville	Addison Wesley	£7.95
The Language of the Dragon	Mike James	Wiley	£6.95
Learning To Use The Dragon 32	George Knight	Gower	£4.95
Load And Go With Your Dragon	Phipps and Tomms	Phipps Associates	£5.95
Make The Most Of Your Dragon 32	Clive Gifford	Interface Publications	£5.95
The MC6809 Cookbook	Carl D. Warren	N/A	£6.95
Programming the 6809	Zaks and Labiak	Sybex	£12.50
Programming the Dragon 32	Peter Lafferty	Newnes Technical Books	£6.95
The Power of the Dragon	Sharp and Botton	Microsource	£5.95
35 Programs for the Dragon 32	Dr Tim Harnell	Softec	£4.95
6809 Assembly Language Programming	Leventhal	Osbourne/McGraw Hill	£12.50
60 Programs for the Dragon 32	Erskin and Walwyn	Pan	£5.95
Software for the Dragon 32	Best of Personal	Computer World	£5.95
The Working Dragon 32	David Lawrence	Sunshine	£5.95

to each of the areas already explained.

The book is suitable for primary school children (although *not* for those who have difficulty with reading) and has a clear text. My criticism of this book is its cost: £4.95 for 56 pages of large print and illustrations seems exhorbitant when compared for example to *Inside the Dragon* which has probably 20 times the text for an extra £3.

Learning to use the Dragon 32 by George Knight is also aimed at young Dragon users. The book is jargon-free and explains simply, for the most part, what you need to do. However, using a full half-page photograph to show a cassette tape and including photographs of a cassette player, five of the Dragon itself and one of a Centronics printer (attached to an Apple II) contributes little to the knowledge of potential buyers.

The next chapter is as bad. This includes large photographs of the screen when the micro is first switched on; when a CLOAD command is being entered; whilst the micro is searching and finally when it gives the OK after loading. I doubt if any of this would be of interest to readers of this magazine and young children would learn far more by being let loose on the keyboard than by ploughing through such a text.

The author then jumps to a mind-blowing explanation of the execution of a program. The task involves printing THE, DOG and SHOW in different combinations on the screen. To explain this he produces a 23-entry diagram showing memory contents at intermediate states of computation. He moves on to string manipulation and to describe some peripheral devices and their use. He unnecessarily instructs readers to OPEN "O", #−2 whenever using the printer and the un-plain English award of the year must go to: "For example CHR$(142 + 112) produces character 142 except that the green area (shown here as black) is orange".

Wasted space

George states that explaining joysticks is too complicated to cover in his book, as presumably is high resolution graphics which only gets two pages and the TIMER feature which he claims does not exist.

If you compare this to the Boots guide, which is also £2 cheaper, it is shown up for the waste of space it is.

The *Boots Guide to the Dragon 32* is a really useful handbook and primer in one. "The hardware of computing", says Ian Sinclair, "consists of all those bits that you can drop and spill coffee over". The first picture is of a cut-away mains plug to help you with your wiring (check yours now!) and immediately helpful suggestions abound. How about a 2-to-1 tv adaptor so that you don't have to continually pull out and re-plug aerials — these are sold as a Panda Pack in DIY shops. What about a four-way socket strip — you'll need all four eventually.

There's even some suggestions on likely hiding places for tuning panels on older-style televisions, some tips on types of tape to use and a reminder to the uninitiated to wind on the plastic leader tape (use a biro). He gives a checklist for playback/recording faults, testing with a four-line program consisting of REMs rather than a game it's taken you hours to type in.

The author points early on to PRINT TAB(x) and multiple TAB statements along with a useful function for centering strings for titles: PRINTTAB(16-LEN(X$)/2); X$ where X$ is the previously defined string you wish to centre. This he frames by using concatenation of strings so that the novice can immediately create pleasing effects on the screen early in his programming career.

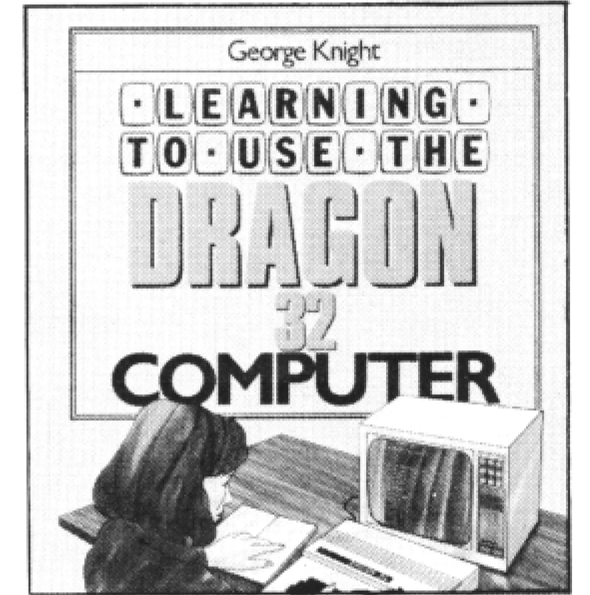
This was one of the only introductory books I have come across which explicitly points out the equivalence of <= with <= in Dragon Basic. Some micros are very particular about this and it adds confidence to the novice programmer not to leave this stone unturned.

The book quite deliberately sets out early on to declare the importance of "mugtraps" in inputting responses. This is a pet theme of mine, for I believe that any program which crashes because you make an inappropriate reply is no good to man or beast. To establish this principle early is good for technical reasons therefore, but it is also of use because it encourages program writers to remember that they write for an audience. That audience might react in any way to their programs, for we are not all the same.

The philosophy behind this handy guide is that you will develop from a program user to a program modifier to a program writer. The author has thus built in a structure to achieve this.

His first moves towards writing programs is to establish firmly the principles of program design. This, he states, starts with the machine switched off and preferably in another room. His presented sequence is where written aims lead to Basic foundations: "Design needs planning and you can't plan properly with the temptation of a keyboard in front of you".

Ever practical, Ian invites the reader to keep one copy of his own programs with all the REM statements intact and store this away somewhere. The "working program" is the version you use with all unnecessary lines stripped out for speed. If you get into difficulties you can then refer to your full copy to search for relevant routines. He gives a useful tip too, in that if you want to test out each stage as you encode it (and who doesn't) a simple Line 1 GOTO 100 will save you constantly reviewing your titles and instructions each time you RUN it. This will be removed at the end too.



Have you ever spent time looking up the numbers for those PRINT @ graphics? Look no further, the Boots guide gives you a simple formula to work out the ones you want. Moving on to hi-res graphics LINE, PSET, BOX & FILL are cleverly introduced without getting lost in cumbersome rules. Sinclair is obviously impressed with the machine: "The graphics capabilities of the Dragon are spectacular . . . most other machines could only do these actions with a lot of very complicated programming". By way of illustration he gives a 14 line program to demonstrate the rotation and scaling of a shape on the screen.

Animation

In his description of animation using the useful GET & PUT, Ian uses Martin Layley's method for working out the dimensions for the array. This is one area where the Dragon manual was seriously wrong so I reproduce the method here to prevent you from wasting the memory space you were led to believe you needed:

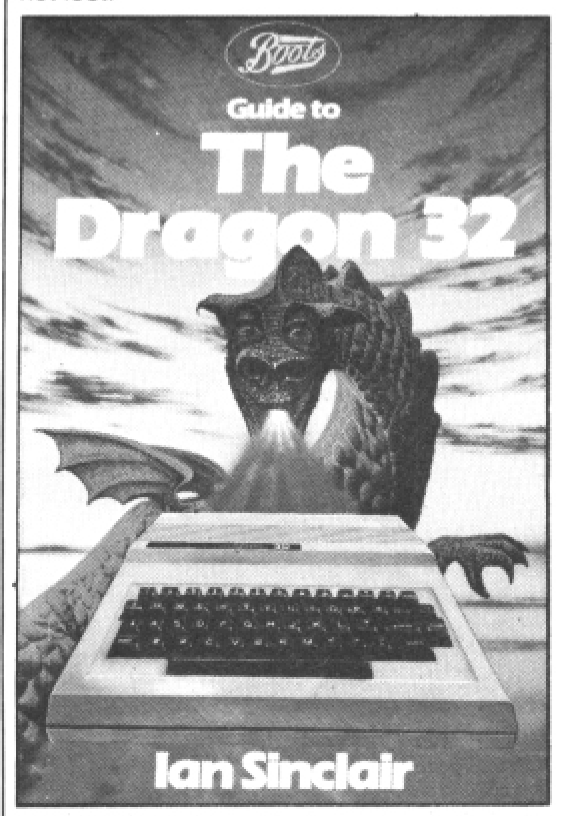
(1) Find the difference between the 'X' numbers of the GET box, and then the 'Y' numbers.
(2) Multiply these differences together and divide by 5, round up if there is a fraction.
(3) Now divide this answer by
  (a) 8 in PMODE 3 or 4
  (b) 16 in PMODE 1 or 2
  (c) 32 in PMODE 0.
  Round up again if the answer is a fraction.
(4) Now use a two dimensional array DIMZ(0,A) where A is the final figure from step 3.
(5) If you get an error message, increase A by 1.

He uses this method to animate his listing "Squids in" which along with a Data processing program he gives at the end of the book for you to use along with what you have learnt. At £2.95 this book published by Granada takes some beating as a value-for-money introductory guide and I thoroughly recommend it.

Penguin books also publish an "indispensable guide to your home computer". Written by David Bannister, *Getting the most from your Dragon 32* is widely available from the larger stores. It too has a section on connecting up the hardware and emphasises the planning process of programming. The author's approach is to get you to break down simple tasks like making a pot of tea or filling a fountain pen into their constituent parts. This idea is then utilised when problems are set asking you to write programs involving the calculation of compound interest and working out the possibility of two people at a party having the same birthday date. This is a most dry approach. It's almost as if the author is afraid that enjoying computing — drawing circles and painting them, or printing dubious messages on screen ▶

◄ — takes away its importance.

This chapter also gives some guidance on saving programs on cassette. Now the tip I was given when I started was to record only one program on each side of a tape (for speed of access to any particular program) and to record each program three times. ClOs are relatively cheap and abundant so for 50p you will have a sure-fire recording of any program. The other thing no book seems to tell you is that if you type in a program and then type CLOAD by mistake or search for your CSAVEd program and cannot find it, all is not lost.



The RESET button will restore your control over the keyboard and the memory will be unaffected. You can then LIST and CSAVE your program. David's section doesn't contain this help.

Generally, the book's illustrations are more helpful than the colour photographs which feature the Dragon itself (in case you've forgotten what it looks like) and various simple screen displays.

For the novice to be faced with chapter 3 "how the computer works inside" is quite ridiculous. This contains such gems as: "There is a conditional jump instruction, saying: 'Jump to the address indicated if the accumulator is zero' . . ." and goes on to describe the instruction register, fetch cycles, "and gates" and "or gates". All this before mentioning variables, loops and data. If ever there was a case for believing that a book had lost its sense of direction, this is it. He later restores the position by including some hints on debugging and a run-down on editing lines.

Another example where the author's knowledge is a hindrance to easy explanation comes in "practice makes perfect", a chapter to help you write programs. He shows readers the way to get random numbers to 100:

LET X = INT(RND(o)*100) + 1,

Computer non-experts like us use

X = RND(100)

Other chapters give listings for a perpetual calendar, a dice game, prime factors and a number sorting routine.

The book goes on to tell readers of the three ways to get sound from your Dragon: playing via your cassette recorder with AUDIO ON: MOTOR ON; SOUND X, Y and the PLAY command. If you've never used this try this little routine to make the Dragon roar:

```
10 INPUT A$
20 PLAY A$
30 GOTO 10
```

Be careful what you put in. Try combinations of just the letters A to G at first.

Penguin's book gives some guidance on computer attachments with ideas on what to look for in joysticks and on choosing a printer. The author's parting shot is to tell his audience where they might get software (called such because of the ease with which the instructions can be changed): "The cheapest of all . . . involves more work and a lot of typing. Magazines . . . contain programs written by users. They are often ingenious and many are better than those commercially available".

Disappointing

It is worth looking at them, he says, just to see how other Dragon owners have coped with certain problems, or got round some of the limitations of the machine. Well, *Dragon User* readers scarcely need to be told that — in fact that's my opinion of the book as a whole. More can be got out of a couple of editions of this magazine and a bit of experimenting than from this, very disappointing book.

A much better proposition for the same price is Brian Lloyd's *Dragon Trainer* which describes itself as a handbook for beginners. Here a disclaimer is needed. *Trainer*, and some other books I look at later, comes from Sunshine — which also publishes this magazine. My only connection with the company is as a freelance writer. I hope that the reservations I raise about their books here will convince readers of my independence, and reassure them that any praise is merited.

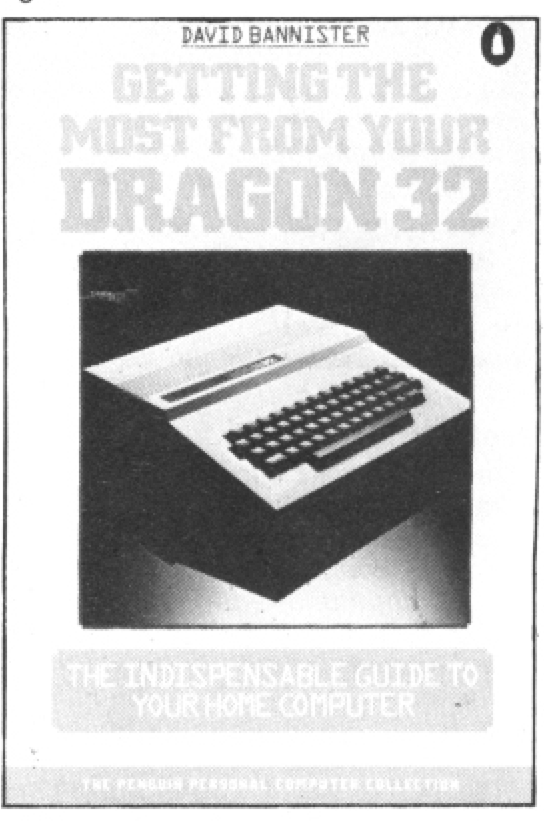
Trainer was written assuming that its readers would have little or no knowledge of computer programming and sets out deliberately to rectify this. The author claims to have tried out each section on complete novices and re-written where necessary in the light of this experience. It certainly gives the feeling of a friendly helper looking over your shoulder and it's difficult to fault its clarity. It is not, however, a book to be dipped into. It needs reading, stage by stage, and in this way the book will take you through the commands as and when you need them so that you can get down to writing your own programs as soon as possible.

Getting started *Trainer* style does not consist of 11 different ways of approaching the on/off switch as we have seen earlier but introduces you to the quirks of the keyboard and inverse video. The PRINT commands and the idea of line numbers are put together with a simple definition: "A variable is a value which can be changed" demonstrated by a simple questionnaire program.

Brian's commitment to confidence building is admirable. Not only does he omit the unnecessary LET X = command so beloved of those who want to impress with the idea of computer mystique (I know something you don't know) but specifically says: "All variables have a value of zero before you use them, it is perfectly alright to refer to a variable which has not yet been given a value".

The author wastes no time in getting down to teaching the powerful IF THEN statement which he also uses to introduce inequalities, a concept very difficult to grasp for those whose schooling 10 years ago or more taught them that equality ($10 \times 4 = 37 + 3$) was all that mattered.

He also rightly makes plain that the opposite to N<10 is N> 9 a point which needs making to novice programmers. This also serves to remind us of the slave nature of the microcomputer. It cannot guess that you mean it to distinguish between values of N less than 10 and those not satisfying this criterion. It only obeys orders. Make sure those orders are right.



The author shows the space saving value of loops by getting readers to type in seven lines like this:

```
10 CLS3
20 INPUT "WHICH MULTIPLICATION
   TABLE WOULD YOU LIKE"; N
30 FOR M = 1 TO 12
40 PRINT N; "*"; M; "="; N*M
50 NEXT M
60 FOR Z = 1 TO 4000: NEXT Z
70 RUN
```

I never understand why so many programs use * to denote multiply in times tables. The symbol is, of course, necessary in Basic for the operation to be carried out, but, in string form 'X' or its video inverse is much clearer.

The book makes a couple of important points which need to be known before incoprorating branching commands into your programs:

(1) Any commands after GOTO com- ►

mand on the same line will not be carried out.

(2) The line numbers after GOTO or GOSUB cannot be replaced with a variable.

Unfortunately, it does not go on to say that we can get around this latter restriction for the most part with on . . . GOTO. This command is, however, dealt with later in the book.

By comparison with the rest of the book the section on tape loading and saving is poor and there is even a mistake in the list of Editor commands ('K' in fact deletes the rest of the line from the current cursor position). Apart from these lapses the book deals well with each Basic word, giving sufficient detail for you to use it with confidence. Every now and then a few are put together to make up a useful routine. PRINTTAB, DEL, RND and RENUM are all covered. This latter I use periodically when program writing to check that my subroutines all connect and to send me helpful UL error messages if not.

The author reminds us that an accidental break can be corrected by typing CONT and that TRON and TROFF are useful tools in error trapping. Unfortunately these last two commands cause the VDG chip to be dedicated to the text screen only, so I find are of little use in debugging graphics programs.

Brian is confidence-building again when he shows a simple mugtrap to cater for answers from "ya" to "yep" for "yes" which would otherwise cause errors:

```
170 IF LEFT$(A$, 1) = "Y" THEN RUN
```

I would also include OR "y" to be really sure.

Repertoire

As you progress through the book and build up a repertoire of Basic, so the lists of programs you are given become more complex. "Breakout", for example, concludes the chapter on graphics and "Artist" the one on PEEK and POKE. This latter allows you to design a shape on the text screen by altering the memory contents in the screen memory addresses from 1024 onwards.

The musical potential of the Dragon has a chapter devoted to it. Try typing in this:

```
T3DDDL2GO + DL5CO - BAO +
  L2GDL5
CO - BAO + L2GDL5CO - BO + CO -
  L3A
```

As you will see and hear, to get more subtle sounds you need more knowledge. The play implications of 0, ›,.,+ and so on are fully discussed.

The philosophy of building confidence appears again in the chapter on hi-resolution graphics. Each of the commands is used and described well. The statement: "The COLOR command is followed by the number of the colour you want to draw in, and the background colour you want" is clear and concise and will stick in the mind. Unfortunately this is followed by "if you wanted to draw in red with a green background you would use the command COLOR3,1" (oops!).

Trainer gives a neat program to introduce you to PCOPYing and another series of formulae to GET and PUT with a one-dimensional array. The book ends by showing you that the 24K of user memory can be increased by 4.5K by typing PCLEAR1 and extended to 30K by POKEing the value 6 into locations 25, 27, 29 and 31.

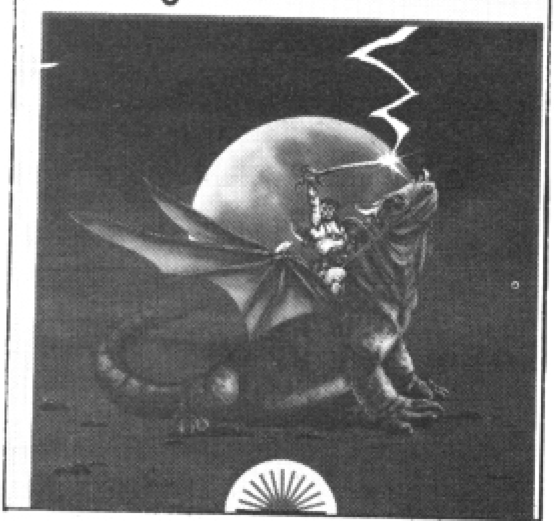
Appendices include listings of "Drawing on the hi-res screen", "Alarm Clock", and "Valley of Death". The latter is a huge graphic adventure game which takes 14 pages to list. For £5.95 I believe *Dragon Trainer* to be the best of this bunch of trainers for those who want an introduction to the Dragon.

If you've owned a Dragon for some time now, or learned the fundamentals on some other machine, you might well be looking for help in putting together programs more effectively. Two books designed to fulfil this need are *The Working Dragon 32* by David Lawrence and *Dragon 32 programmers reference guide* written by John Vander Reyden.

The first of these describes itself as a library of practical subroutines and programs. The author presents a collection of programming modules in each major area. Storing and searching, managing money, drawing, education and that elusive high-resolution text — each has a chapter devoted to it as an example of modular programming. David explains his subroutines within the context of each major program but also with regard to its general application. He points out that when you are writing a number of programs, it is useful to build yourself a library of general-purpose subroutines and select and add to them for individual needs. A useful tip emerges even from his first presented module. This is to have a set line in every program which saves the current version as far as you have developed it. For example:

```
1 GOTO 3
2 CSAVE "UNFILE": SOUND 1, 1: STOP
3 REM
```

Thus if you keep a spare tape in your recorder you can type GOTO2 every now and then putting your hard-worked-for data safely filed away in case of accidental freeze-ups or power failure. He points out that you are far more likely to do this frequently if a simple command is all that's required and will save an awful lot of frustration (don't we all know it) of seeing hours of work lost in a moment.



After reading this I now incorporate it in my own programs but with the addition of POKE HFFD6,0 (I've been caught by that one too). The modules presented in each section have a first-class commentary. A testing routine is given in each case to make sure that errors can be trapped before they interfere with other parts of the program.

Test the tester

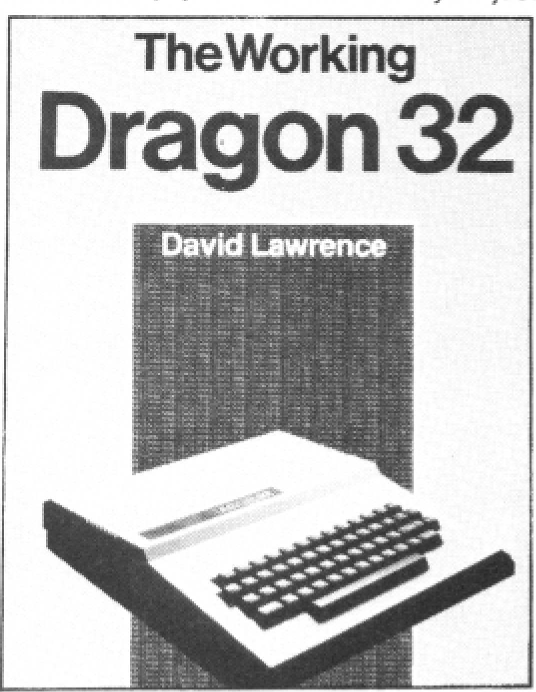
The programs themselves vary in their usefulness. The modular education ones amount to little more than question and answer sessions. In one case this involves the tester drawing items on the screen which the student has to name. This is supposed to teach young children to read. Apart from the fact that it seems to me to be more of a test for the tester to get his shape recognised than the child — it is of course a test of spelling (encoding not decoding). It also cuts across current practice to use the power of computers only for those tasks not better done in other ways — ever heard of Flashcards?

The program "Where" involves the tester drawing a map on the screen and the child has to name the city indicated. Presumably divine inspiration supplies the appropriate names.

A much more useful chapter follows offering solutions to the Dragon's basic flaw, its lack of hi-res text. "Characters" allows you to build up any character capable of being fitted into an area of 32 × 32 picture elements (known as pixels). Once designed these characters are stored on tape for subsequent use in other programs. The author claims that in this way the Dragon's capabilities can be substantially extended. The advantage of this method over the usual DRAWing, he claims, is that there is no need to go through the painfully slow process of building up the fairly complex strings that will be drawn and writing them into each new program. The given modules help you design your characters by means of an on-screen grid and permit movement within the grid, namely rotation and inversion.

Having saved your characters on tape a new program "Dictionary" is built up, again from a series of modules. This collects the data and puts your shapes into memory from where they can be called as you want them. This method is not, of course, restricted to text characters but could be developed for, say, sets of symbols for electronic diagrams, arcade games shapes, chess pieces or the Russian alphabet. The screen dump mixing graphics and text (this one was created using Paul Bernard's Picture Writer), may give you some idea as to the usefulness of such a facility and the characters you might design. ▶

◄ Useful as David's programs are, I think their benefits over DRAWing may be overstated. All this CSAVEing and CLOADing is tempting providence (the I/O fairy is not always kind), and has he never heard of merging programs? The truth lies somewhere in between. If you want a collection of a large number of non-alphabetic symbols then "Character" and "Dictionary" are probably your best tools. If you just want to write on hi-res, then use the strings which have already been worked out for you (often found in magazines). You only need type them out once, save this program on tape with the routine at 10,000 and then by the use of DEL, RENUM, PEEK and POKE you will be able to put them into any program.



The author puts his techniques from other chapters to use in a variety of utilities loosely titled "Handy Programs". He lists a database "Name and Number" suitable to hold facts about calorific value of foodstuffs or costs of items in stock. His program "Typist" consists of a number of modules which together aim to turn you into a touch typist and further routines are suggested to involve words per minute and other refinements.

Subroutines in "Texted", a simple word processor program, may be of use to those who own, or hope to own, a printer. "Texted" has a screen editing facility. "Music" helps you compose tunes of your own and allows you to save data on tape to use in programs of your own. "Graph" is a graph-drawing tool. You can draw line graphs of a variety of data, specifying the units and the set-up of the axes. It utilises text generated by "Character", discussed earlier.

These substantial programs, like the rest of the book, are well commented on, and the subroutines can form the basis of a library to be incorporated into almost any program you care to write. If any I have described strike a chord, then perhaps *The Working Dragon 32* is the book for you.

The Dragon 32 programmers reference guide goes for the same ground in that it aims to make you better, rather than get you started. It is organised as a reference source for both Basic and experienced 6809 machine language programmers. Like most reference books, your ability to apply the information depends on your knowledge of the subject. In this case if you are a novice Dragon programmer the facts and figures in the book will not be of as much use to you as to those with greater knowledge and experience.

The book begins with a complete Basic dictionary of statements and functions, a detailed description of each word and examples on how to use it. It even gives the average time taken to execute. For example:

Motor
- Turn the cassette motor on or off
- MOTOR ON
  MOTOR OFF
- Allows the motor of the cassette to be controlled by a program for creating special effects (see AUDIO)
- MOTOR ON 0.5272 secs
  MOTOR OFF 0.0005 secs

There are also some interesting details on decimal, hexadecimal and octal numbers and an error in the Basic. Try this:

```
10 X = 53.74 : Y = 51 + 2.74
20 PRINT X,Y
30 IF X = Y PRINT "RIGHT" ELSE
   PRINT "WRONG"
```

For different numbers the above equation will give 'RIGHT' or 'WRONG'. For example, I found that for X = 70.08 and Y = 1.1 + 0.9 + 70.08 the equality was accepted.

Strings

There seems no set pattern as to how the decimal representation is affected in floating point addition. Having identified the problem, the author gives us the solution. Converting the numbers into their string equivalents using STR$, Basic then recognises their equality.

Chapter 2 deals with graphics starting with a discussion on the quality of the picture resolution. Five semigraphic modes and eight true graphic modes are detailed and possible applications given. For instance:

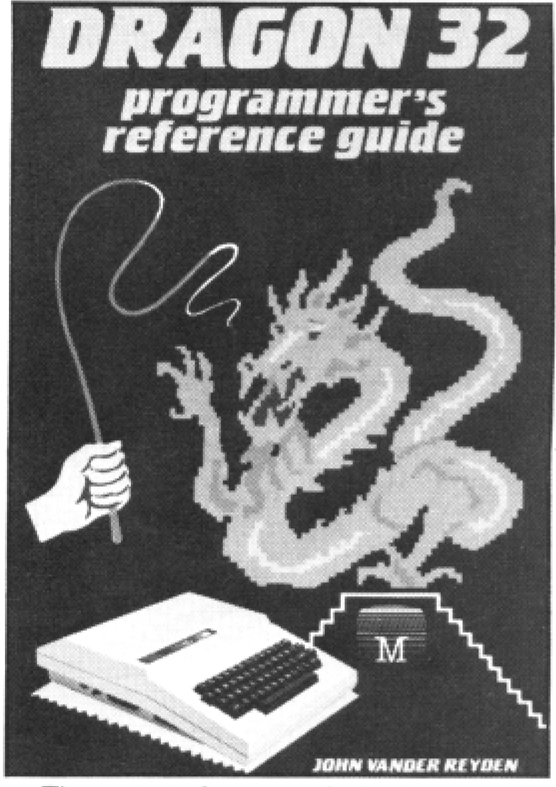
Semi-graphic 6-24 — for higher resolution in the vertical axis; could be good for accurate bar charts but can be wasteful in terms of memory.

As only five of these modes can be reached through Basic, he gives the POKEs needed to select each of the others. When memory locations between 65472 and 65477 have been set the problem becomes how to fake commands such as LINE, DRAW and PAINT. Here you're on your own although the author does give some tips.

In "Sound", John devotes some space to the PLAY command and gives listings for you to play "God Save the Queen" and "In An English Country Garden"! He then, more interestingly, goes on to assembly language giving a routine for setting up the PIA registers for the production of sound. There is a distinct advantage in using machine code in this area. When using sound in a Basic games program you have to keep the noises short as the processor is tied up in producing the sound and the program has to wait until it has finished.

In machine language programs, however, you can do some processing in between the toggling of the speaker and so longer notes can be played without disturbing the flow of the game. Thus you should be able to reproduce "Bolero" with your Torville and Dean skating game (or Bo Derek game as takes your fancy).

A fascinating machine language routine within a Basic program is given which will teach your Dragon to speak. When run the menu gives a variety of choices allowing you to digitally encode a few seconds of speech and then analyse it graphically or save the data on tape. Your voice (or music) is entered via the cassette system, either previously recorded or direct. You can check on the quality of the coding before you save to tape by reproducing the sounds presently held in memory. In my case the reproduction wasn't very good (about the same as "Android Attack"). This might be because my microphone/cassette system wasn't up to the job, the volume controls were not set right or maybe this method is not up to much anyway. It nonetheless gave hours of fun to the family (ever tried to get a cat to "miaow" on cue?). It was well worth the 40 minutes it took to type in. Mind you, this wasn't the first time we had tried to enter that program.



The worst feature of these machine code within Basic programs is that DATA errors are not easily spotted and can lead to disastrous results. These latter books are full of them and so now are the magazines, so if you go for one of these, here is the voice of experience with a few tips to avoid calamity. Firstly double check the data, especially the hexadecimal addresses. Secondly save a copy *before* running it — then at least you will have an albeit imperfect copy safe on tape should the computer freeze or play up because of a careless POKE. Lastly, count the pieces of data and do a little dummy run on your program like this:

```
1 READ Z:X = Xt1
2 IF Z = 999 THEN PRINT X - 1;
  "PIECES OF DATA"; :STOP
3 GOTO 1
```

►

◄ 10000 DATA 999

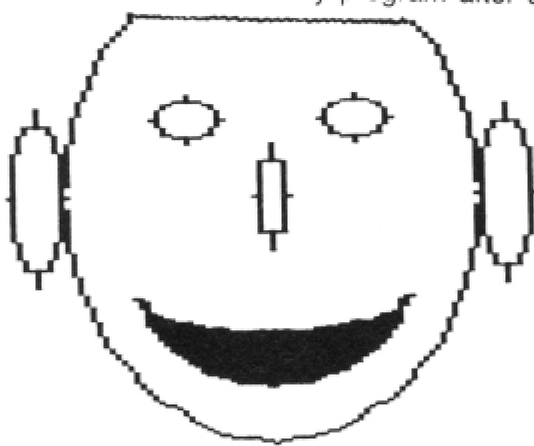
Program writers could help in this too by incorporating a data check in the list:

```
1 READ Z:IF Z = 999 THEN GOTO 5
2 X = X + Z: GOTO 1
5 IF X = 29743 (or whatever is the
  correct value of the sum of the data)
  THEN GOTO 10
6 PRINT "DATA-ERROR"; :STOP
10 REM ** START OF PROGRAM
   PROPER
```

Now that we have the facility to reproduce the human speech thanks to the *Reference Guide* what can we do with it? In the first instance it will almost certainly not be clear enough to hold the instructions for a game. It won't be long enough either, for the stored speech takes up to 6K of memory and will last about one to four seconds depending on content. You could use it, however, to give short, often repeated commands like "Fire" or "Go" at the start of a race game, or "Good", "Well done" and so on in a test. The book gives the exact instructions as to saving machine code and data to call from within your own programs.

Chapter 4 gives a gentle introduction to machine code — enough to get you interested but not enough to get you programming. There is listed for you a machine code monitor with which you can enter, modify and display parts of memory, as well as find a string of characters within the memory. It will execute an assembly language program and convert numbers from hexadecimal to decimal and vice-versa. Later in this chapter the author provides a summary of handy ROM routines which can easily be used in machine code language programs.

This excellent book ends with a superb final chapter on handy tips and routines. After a discussion on the machine implications on speeding things up a variety of short subroutines are given. To disable the "break" key, for example, John POKEs locations 411 to 415 with the values 228, 203, 4, 237 and 228. The break key is then turned off by POKEing 410 with 236 and on with 57. Unfortunately this must be used directly from the keyboard but he does list a Basic program to create a machine code file which you can call up from within programs of your own and turn the break key off and on at will. Other paragraphs show how to use set/reset in semi-graphic modes, create an auto-key repeat, allow the Dragon to read two keys at once and recover any program after a NEW command.



Simple program from the Reference guide

Apparently NEW does not wipe out a program, it just modifies the Basic pointers so that the program cannot be assessed. John's program will create a machine code file which you can reload into your micro should you ever inadvertently have NEWed your latest masterpiece. He even gives the POKEs which you could use directly from the keyboard should you not yet have made the machine code file but need its assistance.

Two other sections deal with redefining Basic keyboards and their actions. I've always fancied a machine which started up on "JOG" to keep up with the latest fashion. I can do it now. The book also gives a merge routine and some suggestions as to its usefulness. Some graphics hints are made. Page swapping to demonstrate animation is very effective and a circles program shows the power and versatility of this command.

The *Programmers reference guide* at £6.95 seems much the better to me of these two "second level" books. Its routines seem more general and tips more pertinent to the sort of programs I am likely to write. You do not need to know about machine code to use some fast and useful files. John understands too that these tools are only liable to be of any use in your own programs, so explicitly tells you how to arrange this.

That's it for this month — more in subsequent issues looking at the books that will further expand your programming capabilities. ■

OPEN FILE FOR DRAGON USERS

Send us your Dragon programs, beginning with a general description and then explaining how the program is constructed. Take care that the listings are all bug-free, enclosing a cassette and, if possible, a printout. We pay £6 for each bug-free program published, double for the program of the month. If you have any problems with the listings, please send your queries to the appropriate author, *Dragon User*, 12-13 Little Newport Street, London WC2R 3LD.

Tic, tac, toe

From Helen Johnstone in Suffolk

THIS PROGRAM, which is more commonly known as "stone, paper, scissors", is a game for one player. The players chooses either "scissors", "stone" or "paper", and then waits for the Dragon to choose one randomly.

If stone and scissors are chosen, stone wins as it blunts the scissors. If stone and paper are chosen, paper wins as it wraps up the stone; and if paper and scissors are chosen, scissors claims victory by cutting the paper.

A match is won by gaining three points, one point is achieved by winning one try. A game is won by gaining three matches.

Program notes

Lines	
50-160	Defines graphic strings.
170-220	Draws and plays title sequence.
230-290	Instructions.
300-320	Alters resolution according to type of television.
330-360	Initialise variables.
370-400	Draws count-down numbers.
410-460	Computer chooses item, checks for wrong inputs.
470-520	Draws out chosen items.
530-620	Checks for winner and prints WIN.
630-700	Checks for game or match wins.
710-740	Checks for re-run.

Variables

T$,C$	Title (T and C) graphics.
H$(=3), W$(=2), O$(=1)	Countdown numbers.
A$, B$	Scissors graphics.
P$, D$	Paper graphics.
E$	Stone graphics.
Y$	Draw graphics.
Z$	Win graphics.
U	PMODE variable.
RD,DR	Dragon score.
RP,PE	Player score.

```
10 ' TIC TAC TOE  HELEN JOHNSTONE
                  FEB '84
20 POKE65495,0
30 CLEAR 300
40 PMODE4,1:SCREEN1,1:PCLS
50 CLS:T$="R30D4L13D26L4U26L13U4"
60 C$="R30D4L26D22R26D4L30U30"
70 H$="BM118,40;R20D30L20U4R16U9L1
6U4R16U9L16U4"
80 W$="BM118,40;D4R16D9L16D17R20U4
L16U9R16U17L20"
90 O$="BM118,45;M126,40;R4D26R8D4L
20U4R8U22;M118,49;U4"
100 A$="BM50,50;D70M58,75M50,50L10
D15R10D10M40,118M70,54M78,58M72,70
M64,66M70,54M50,74"
110 B$="BM200,50;D70;M208,75;M200,
50;L10D15R10D10;M190,118;M220,54;M
228,58;M222,70;M214,66;M220,54;M20
0,74"
120 P$="R70D70L80U60R10U10"
130 D$="D15U10R10U5L10BR15D15U10R1
0D10U15L10BR15D15U10R10U5L10BR15R1
0L10D5R10L10D10R10BR5U15R10D5L10R7
D10"
140 E$="R12D4L8D9R8D17L12U4R8U9L8U
17BR14D4R4D26R4U26R4U4L12BR14R12D3
0L12U30R4BD4R4D22L4U22BU4BR10D30R4
U26R4D26R4U30L12BR14D30R12U4L8U9R8
U4L8U9R8U4L12"
```

continued on page 47

```
150 Y$="BM82,140D30R20U30L20R4BD4D
22R12U22L12BU4BR20D30R4U10R10D10R4
U14L13U2R16U14L20R4BD4D4R12U4L12BU
4BR20D30R4U10R12D10R4U30L20R4BD4R1
2D10L12U10BU4BR20D30R20U30L4D26L4U
26L4D26L4U26L4"
160 Z$="D30R20U30L4D26L4U26L4D26L4
U26L4BR26D4R8D22L8D4R20U4L8U22R8U4
L20BR26D30R4U26R12D26R4U30L20BR26D
17R16D9L16D4R20U17L16U9R16U4L20"
170 DRAW"BM240,180;D4U2R2U2D4":DRA
W"BM240,178;R8L3D8L5"
180 DRAW"BM80,40"+T$:PLAY"T5001CCD
DEEFFGGAABB02CC"
190 DRAW"BM126,40;R4D30L4U30":DRAW
"BM80,80"+T$:PLAY"DDEEFFGGAABB03CC
"
200 DRAW"BM146,40"+C$:DRAW"BM122,8
0;R12D30L4U11L4D11L4U30R4BD4R4D9L4
U9":DRAW"BM80,120"+T$:PLAY"03CCDDE
EFFGGAABB04CCCC"
210 DRAW"BM146,80"+C$:DRAW"BM122,1
20;R12D30L12U30R4BD4R4D22L4U22":PL
AY"04CC03BBAAGGFFEEDDCC"
220 DRAW"BM146,120;R30D4L26D9R26D4
L26D9R26D4L30U30":PLAY"03CC02BBAAG
GFFEEDDCC01CCBBAAGGFFEEDDCCCC"
230 PRINT@10,"INSTRUCTIONS
              **************":PRINT:PR
INT:PRINT" THE RULES ARE VERY SIMP
```

```
LE.YOU  HAVE TO CHOOSE ONE OF THE
      FOLLOWING WHEN THE NUMBER '1
    COMES ON THE SCREEN.
          '1'=SCISSORS
       '2'=PAPER
240 PRINT@326,"'3'=STONE
       PRESS A KEY TO CONTINUE"
250 ANS$=INKEY$:IF ANS$=""THEN 250
260 PRINT@30,"
              THE DRAGON ALSO CHOO
SES ONE.    SCISSORS CAN CUT PAPER
 BUT NOT  STONE.PAPER CAN WRAP STO
NE BUT  IS CUT BY SCISSORS ETC.THE
 FIRSTTO WIN 3 MATCHES WINS THE GA
ME. DRAWS DO NOT COUNT."
270 ANS$=INKEY$:IF ANS$=""THEN 270
280 PRINT@30,"
              YOUR ITEM IS DISPLAY
ED ON THE   RIGHT SIDE OF THE SCRE
EN, THE   DRAGON'S ON THE LEFT.":P
RINT:PRINT:PRINT
290 ANS$=INKEY$:IF ANS$=""THEN290
300 CLS:PRINT:PRINT"ARE YOU USING
A BLACK AND WHITE T.V.?"
310 ANS$=INKEY$:IF ANS$=""THEN 310
320 IF ANS$="Y" THEN U=4 ELSE U=3
330 CLS:PRINT@66,"WHO AM I GOING T
O PLAY WITH?":INPUT I$
340 PRINT:PRINT"HELLO ";I$;" ARE Y
OU READY.......     THEN LETS PLAY.
":FOR I=1 TO 1000:NEXT I
350 DR=0:PE=0
360 RD=0:RP=0
370 PMODE U,1:SCREEN1,0:PCLS
380 DRAW H$:SOUND190,5:PCLS
390 DRAW W$:SOUND180,5:PCLS
400 DRAW O$:SOUND170,10
410 Y=0:Z=0:D=0:P=0
420 C=RND(3)
430 Q$=INKEY$:IF Q$=""THEN 430
440 A=VAL(Q$)
450 IF A>3 OR A<1 THEN CLS1:PRINT@
230,"WRONG KEYS..TRY AGAIN":SOUND1
20,5:SOUND90,5:FOR I=1 TO 100:NEXT
 I:GOTO360
460 PMODEU,1:PCLS:COLOR 3,1:SCREEN
1,0
470 IF C=1 THEN DRAW A$
480 IF C=2 THEN DRAW "BM20,40"+P$+
"M10,50"+"BM16,60"+D$
490 IF C=3 THEN DRAW "BM25,40;R45M
80,50D45M70,105L45M5,95U45M25,40D1
5BD50R30U10BU55R15D20BM7,57"+E$
500 IF A=1 THEN DRAW"BM166,50"+B$
510 IF A=2 THEN DRAW "BM166,40"+P$
+"M156,50"+"BM162,60"+D$
520 IF A=3 THEN DRAW "BM175,40R45M
230,50D45M220,105L45M155,95U45M175
,40D15BD50R30U10BU55R15D20BM157,57
"+E$
530 IF C=A THEN DRAW Y$:SOUND 96,1
0
540 IF C=1 AND A=3 THEN P=1:SOUND1
,1
550 IF C=1 AND A=2 THEN D=1
560 IF C=2 AND A=3 THEN D=1
570 IF C=2 AND A=1 THEN P=1
580 IF C=3 AND A=2 THEN P=1
590 IF C=3 AND A=1 THEN D=1
600 IF D=1 THEN DRAW"BM5,150"+Z$:S
OUND 120,5:SOUND90,5
610 IF P=1 THEN DRAW"BM150,150"+Z$
:PLAY"T50CCDDEEFFGGAABBO3CCDDEEFFG
GAABBO3CC"
620 FOR I=1 TO 1000:NEXT I:PCLS
630 RD=RD+D:RP=RP+P
640 IF RD=>3 OR RP=>3 THEN 650 ELS
E 370
650 CLS:PRINT:PRINT:IF RD>RP THEN
 PRINT "DRAGON WINS ";RD;" MATCHES
 TO ";RP ELSE PRINT I$" WINS ";RP;
" MATCHES TO ";RD
660 IF RD>RP THEN Y=1 ELSE Z=1
670 FOR I=1 TO 3000:NEXT I
680 DR=DR+Y:PE=PE+Z
690 IF DR=>3 OR PE=>3 THEN 700 ELS
E 360
700 CLS:PRINT:IF DR>PE THEN PRINT"
DRAGON WINS THIS GAME
            ";DR;":";PE  ELSE PRINT"
YOU HAVE BEATEN THE DRAGON
            ";PE;":";DR:PLA$="T100DG
CCDEFGFGCCDEFG":FOR I=1 TO 4:PLAY
PLA$:NEXT I
710 PRINT@140,I$:PRINT"DO YOU WANT
 TO PLAY AGAIN (Y/N)"
720 A$=INKEY$:IF A$=""THEN720
730 IF A$="N" THEN 740 ELSE CLS:GO
TO330
740 CLS:POKE65494,0:END
```

Hunt the Dragon

From Roger Reading in the West Midlands

THIS PROGRAM is based on the children's game of hide and seek, when the player is given the hint "cold", "warm", or "hot", according to how close the player is to the hidden object.

The computer will hide a "dragon", and you have to find it by entering a letter and then a number to the computer's prompt.

The computer will indicate how close you are by showing your chosen square as blue (cold), orange (warm) or red (hot). This game has the added attraction of helping to teach co-ordinates in the conventional x-axis, y-axis format.

Program notes

Lines	
30-100	Set up the initial screen.
130-230	Draws the grid.
240-290	Labels the axes.
300	Chooses a random square within the grid.
330-400	Accepts a letter and then a number input within the grid limits.
410-420	Converts input letter and ►

◄ number to a position on the PRINT @ grid.

430-520 Checks to see how close the player's chosen square is to the computer's chosen square, and then prints the player's chosen square in the appropriate colour.

530-550 Win routine.

```
10 ' "DRAGON"
20 ' R.K.READING
30 CLEAR 1000
40 CLS
50 PRINT @ 72,"hunt the dragon"
60 PRINT @ 128,STRING$(32,"*")
70 PRINT @ 200,"THERE IS A DRAGON
HIDING ON THE GRID. FIND HIM BY
    ENTERING A LETTER THEN A NUMBE
R.THE COLOURS SHOW HOW HOT YOU ARE
"
80 A$="O3L2CL4CL2CCO2L1BG"
90 PLAY "T6"+A$
100 PRINT @ 356,"PRESS ANY KEY TO
PLAY"
110 B$ = INKEY$
120 IF B$="" THEN 110
130 CLS
140 FOR N = 139 TO 149 STEP 2
150 PRINT @ N,CHR$(143+64);CHR$(14
3+80)
160 FOR M = 2 TO 6 STEP 2
170 PRINT @ N+32*M,CHR$(143+64);CH
R$(143+80)
180 NEXT M,N
190 FOR I = 171 TO 181 STEP 2
200 PRINT @ I,CHR$(143+80);CHR$(14
3+64)
210 FOR J = 2 TO 4 STEP 2
220 PRINT @ I+32*J,CHR$(143+80);CH
R$(143+64)
230 NEXT J,I
240 FOR K = 65 TO 76
250 PRINT @ 298 + K,CHR$(K)
260 NEXT K
270 FOR L= 1 TO 7
280 PRINT@ 360-32*L,L;
290 NEXT L
300 X = (RND(10)+139)+(RND(5)*32)
310 RESTORE
320 PRINT @ 448," "
330 PRINT @ 420,"WHICH LETTER DO Y
OU WANT?"
340 INPUT A$
350 IF A$ > "L" THEN 320
360 PRINT@420," "
370 PRINT@420,"WHICH NUMBER DO YOU
 WANT"
380 INPUT B
390 IF B < 1 OR B > 7 THEN 360
400 PRINT @ 420," "
410 Z = ASC(A$)+266
420 Y=Z-(32*(B-1))
430 IF Y = X THEN CLS:PRINT @ 234,
"GOT HIM":GOSUB 530:GOTO 140
440 FOR R = 1 TO 8
450 READ A:IF Y=X+A THEN PRINT @ Y
,CHR$(143+48);:GOTO 310
460 NEXT R
470 DATA -33,-32,-31,-1,1,31,32,33
480 FOR S = 1 TO 16
490 READ C:IF Y = X+C THEN PRINT @
 Y,CHR$(143+112);: GOTO 310
500 NEXT S
510 DATA 66,65,64,63,62,34,30,2,-2
,-30,-34,-62,-63,-64,-65,-66
520 PRINT @ Y,CHR$(143 + 32);:GOTO
 310
530 B$="O3L2CDL1EDCO2AGFGO3CCC"
-40 PLAY "T15"+B$
    -TURN
```

Maths

From Stanley White in South Yorkshire

AFTER SEEING the Maths program in the November issue of *Dragon User*, I decided to develop my own times table program.

Program notes

Lines	
30-470	Set up character arrays.
500-670	Draw board.
1000-1060	Information box.
1070-1080	Clear board arrays.
1100-1140	Define section.
1150-1180	Picks random number.
1200-1510	Draw equation and check if right or wrong.
1600-1680	Clears wrong answer.
2030-2100	Ends, draws score and asks if another game required.
2500-2600	Allows digit inputs and draws on screen.
3000-3010	Draws letters onto screen.
4000-4010	Draws numbers onto screen.
5000-5230	Instructions.

```
1 GOTO3
2 POKE65494,0:MOTOR ON:AUDIO ON:FO
R D=1TO5000:NEXT:CSAVE"X-TABLES":S
OUND1,1:MOTOR ON:FOR D=1TO5000:NEX
T:MOTOR OFF:STOP
3 REM***************************
4 REM*******stan*white*********
5 REM***************************
10 PMODE4:COLOR0,1:PCLS:POKE65495,
0
20 TIMER=0:GOSUB5000
30 REM*set*up*character*array*
40 DIM AZ$(27):FOR T=0 TO 27:READ
AZ$(T):NEXT T
50 DIM N1$(11):FOR T=0 TO 11:READ
N1$(T):NEXT T
60 REM**letters*az$**
70 DATA BU4R4D2L2D2BR2
80 DATA BR4
90 DATA NRU4R4D3NL3D
100 DATA NR4UNRUNR2U2R3D2RD2
110 DATA NR4UNRUNRU2R4DBD2D
120 DATA RNR2NU2LU4R3FD2BD
130 DATA NR4UNRUNR4U2R4BD4
```

continued on page 53

'hands off the Ultradrive!'



At last a computer controlled storage device that will load a 16k program in under 15 seconds* and doesn't use any of your computer's precious memory!

Each ULTRA DRIVE Micro cassette has an enormous 200k byte capacity*.

The ULTRA DRIVE comes complete with all necessary interfaces, its own sophisticated filing system, and a utility cassette.

All this for only
£79.95 inc. VAT
NO HIDDEN EXTRAS

This incredible machine is available for
DRAGON ELECTRON BBC
NASCOM ORIC TANDY
COMMODORE

IKON
certified digital mini-cassette
A

*For the Dragon computer. Specifications may vary sightly for other computers.

10 day money back guarantee
Purchase price will be refunded if not delighted

IKON
COMPUTER PRODUCTS
IKON COMPUTER PRODUCTS, DEPT. E, KILN LAKE, LAUGHARNE, DYFED SA33 4QE (Tel. 099 421 515)

To: IKON COMPUTER PRODUCTS, DEPT. E, KILN LAKE, LAUGHARNE, DYFED SA33 4QE (Tel. 099 421 515)
Please send me ______ ULTRA DRIVE(s) for the ______ computer @ £79.95 + £3.45 p&p each
______ Data certified cassettes @ £4.03 each. ______ Boxes of cassettes @ £20.13 per box of 6
______ Cleaning cassettes @ £4.60 each. I enclose cheque/P.O. value £
Please charge my Access/Visa card no . . . . (Delete whichever not applicable)
Name:
Address:
Signature:
(Please allow 28 days for delivery. Remittances will not be cashed until goods are ready to be despatched)
*For the Dragon computer. Specifications may vary slightly for other computers.

```
140 DATA RNULU2NR4U2R4BD4
150 DATA RNR3NU2LU4R4BD2D2
160 DATA RNULU2NR3U2BR4D4
170 DATA BR2NU2RNU4BR
180 DATA R2NU2RU4NL3RBD4
190 DATA NU4RU2R2NU2RD2
200 DATA NU4RNU2R3
210 DATA U4R2ND3R2D4
220 DATA U4RF3NU3D
230 DATA RNU2LU4R4D4NL4
240 DATA NRUNR4U3R4D3BD
250 DATA U4R4D3NL2DNL4
260 DATA RNU2LU4R4D2L2F2
270 DATA R3NURU2L4U2R4BD4
280 DATA BR3NU2LU4NL2R2BD4
290 DATA NU4RNU2R3NU4
300 DATA U4RD2BD2REUEUBD4
310 DATA NU4R2NU4R2NU4
320 DATA NU2RU2NU2R2NU2RD2
330 DATA BU2U2RD2R3NU2D2
340 DATA U2RND2R3U2NL4BD4NL2
350 REM**numbers*n1$**
360 DATA U6R3D6NL3
370 DATA BRNU6BR
380 DATA U3R3U3NL3BD6NL3
390 DATA R3U3NL3U3NL3BD6
400 DATA BU3NU3R3NU3D3
410 DATA R3U3L3U3R3BD6
420 DATA NU2R3U3L3U3R3BD6
430 DATA BU6R3D6
440 DATA U6R3D3NL3D3NL3
450 DATA R3U6L3D3R3BD3
460 DATA E2H2E2BD6
470 DATA U1BU2U2R3U3NL3BD8
500 REM**draw*table**
510 DRAW"BM0,0;BR4BD8"+N1$(10)+"BR
3"+N1$(10)+"BR3"+N1$(10)+"BR5BD2U1
0L21D10R21"
520 DIM B(12,12):GET(0,0)-(21,10),
B,G:PCLS
530 FOR X=0TO231STEP21:FOR Y=0TO11
0STEP10
540 PUT(X,Y)-(X+21,Y+10),B,PSET
550 NEXT Y,X
560 FORX=1TO234STEP21:LINE(X,1)-(X
+19,9),PRESET,BF:NEXT:'**clear*top
*boxes**
570 FORY=1TO111STEP10:LINE(1,Y)-(2
0,Y+8),PRESET,BF:NEXT:'**clear*sid
e*boxes**
580 S=1:T=0:REM**put*1-12*in*top*b
oxes**
590 FOR X=9TO240STEP21
600 IF S=10 THEN X=X-3
610 DRAW"BM"+STR$(X)+",8":I$=STR$(
S):GOSUB4000
620 S=S+1:NEXT X
630 X=9:S=2:T=0:REM*put*1-12*in*si
de*boxes**
640 FOR Y=18TO118STEP10
650 IF S=10 THEN X=6
660 DRAW"BM"+STR$(X)+","+STR$(Y):I
$=STR$(S):GOSUB4000
670 S=S+1:NEXT Y
1000 REM**draw*information*box**
1010 LINE(0,122)-(252,190),PSET,B
1020 DRAW"BM48,130":A$="LEARN@YOUR
@TIMES@TABLES":GOSUB3000
1030 DRAW"BM44,175":A$="YOU@HAVE@U
PTO@THREE@TRYS":GOSUB3000
1040 DRAW"BM80,185":A$="PRESS@E@TO
@END":GOSUB3000
1050 DRAW"BM10,145":A$="TRYS":GOSU
B3000
1060 DRAW"BM200,145":A$="SECTION":
GOSUB3000:DRAW"BM222,160"+N1$(Z)
1070 REM**clear*arrays**
1080 SCREEN1,1:DIM C(12,12):FOR X=
2TO12:FORY=2TO12:C(X,Y)=0:NEXTY,X:
K=0:SC=0
1100 REM**define*section**
1110 IF ZZ=4 THEN Z=RND(3)
1120 IF Z=1 THEN Z$="H"
1130 IF Z=2 THEN Z$="I"
1140 IF Z=3 THEN Z$="J"
1150 REM**pick*rnd*numbers**
1155 KC=0
1160 I=RND(11)+1:J=RND(11)+1:H=I*J
:KC=KC+1
1170 IF C(I,J)=1 AND K>118 AND KC=
1 THEN LINE (1,160)-(200,150),PRES
ET,BF:DRAW"BM88,160":A$="PLEASE@WA
IT":GOSUB3000
1180 IF C(I,J)=1THEN 1160
1200 REM**clear*ans*box**
1210 LINE(1,160)-(200,150),PRESET,
BF:C=0
1220 REM**draw*numbers**
1230 IF Z$="I"THEN GOTO1260
1240 I$=STR$(I):IF LEN(I$)<3 THEN
X=94 ELSE X=88
1250 DRAW"BM"+STR$(X)+",160":GOSUB
4000
1260 DRAW"BM102,160;E5BL5F5":REM**
x**
1270 IF Z$="J"THEN 1290
1280 I$=STR$(J):DRAW"BM112,160":GO
SUB4000
1290 DRAW"BM120,160BR6BU4NR4BD2R4B
D2BR6":REM**=**
1300 IF Z$="H"THEN 1320
1310 I$=STR$(H):DRAW"BM135,160":GO
SUB4000
1320 REM**draw*ans**
1330 IF Z$="I"THEN DRAW"BM88,160"
1340 IF Z$="J"THEN DRAW"BM112,160"
1350 IF Z$="H"THEN DRAW"BM135,160"
1360 GOSUB2500:C=C+1
1370 LINE(20,150)-(25,160),PRESET,
BF:DRAW"BM20,160"+N1$(C)
1380 DRAW"BM102,145"
1390 IF Z$="I"THEN IF V=I THEN 142
0 ELSE 1480
```
continued on page 55
```
1400 IF Z$="J"THEN IF V=J THEN 142
```



★ CLASSIFIEDS ★

- ★ *Popular Computing Weekly* was voted magazine of the year by the CTA.
- ★ It is Britain's only weekly home computer magazine with a published Audit Bureau of Circulation sales figure.
- ★ It has a classified section that is now required reading for everyone with an interest in small micros, or who wants to buy or sell: SOFTWARE ★ UTILITIES ★ ACCESSORIES ★ SERVICES ★ HARDWARE ★ HIRING ★ CLUBS ★ RECRUITMENT ★
- ★ £5 per SCC semi-display, 20p per word lineage.
- ★ 6 day copy date.

CALL DIANE DAVIS ON 01-437 4343 FOR AN IMMEDIATE QUOTE.



Popular Computing Weekly, Sunshine, 12/13 Little Newport Street, London WC2R 3LD





MONSTER SOFTWARE CLUB

SOFTWARE LIBRARY FOR THE DRAGON 32

- ★ **Software for hire from 11 manufacturers**
- ★ **Over 90 titles to choose from, and growing**
- ★ **TWO YEARS membership for only £8.00**
- ★ **Same-day service**

Software at 10% discount. Send SAE for details and list of titles available to:

MICROBYTE COMPUTER SHOP

19a Lower Warrengate
Wakefield WF1 1SA

MIDDLESEX DRAGON CENTRE

DRAGON 32 COMPUTER	£168.00
DRAGON 64 COMPUTER	£220.00
DRAGON DISK DRIVE	£270.00
DRAGON JOYSTICKS (pair)	£16.95
64 OS9 OPERATING SYSTEM	£39.95
64 OS9 PROGRAMMERS MANUAL	£19.95
64 RECORD MANAGEMENT SYST.	£54.95
STOCK RECORDING	£49.95
64 EDITOR ASSEMBLER DEBUG	£49.95
64 C COMPILER + Manual	£79.95
64 PASCAL + Manual	£79.95
64 DYNACALC + Manual	£59.95
64 STYLOGRAPH WORD PROCESSOR + Spellcheck, mailmerge etc	£79.95
64 CASH & VAT	£49.95

Plus!! Games, Educational and Business Software, Dust Covers, Monitors, Floppy Discs, Cassettes, Paper, Spare Leads, Monitor Stands and everything for the Dragon owner.

★Open 6 days per week ★Reliable worldwide mail order service
★Approved Dragon dealers, service and repair centre

SCREENS MICROCOMPUTERS

6 Main Avenue, Moor Park, Northwood, Middlesex
Tel: Northwood 20664

Access ★ Barclaycard ★ Amex ★ Diners Club ★ Instant Credit

Computasolve Ltd.

8 Central Parade, St Mark's Hill
Surbiton

Telephone: 01-390 5135. Opening hours: 9.30—6.30, Mon—Sat

Dragons from stock	£175.00
Telewriter word processor for the Dragon	£49.95
CTI CP80 dot matrix printer	£299.00
Seikosha GP 100A matrix printer	£240.00

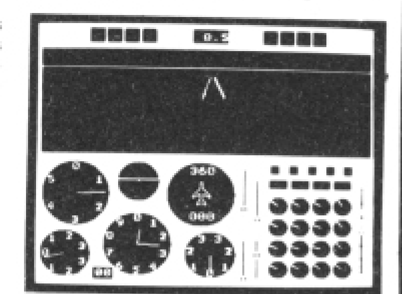
Plus a full selection of software for the Dragon including the latest Microdeal releases

All prices including VAT.

DRAGON/32 BBC MODEL/B ATARI 400/800 TRS80 C/C 32K ELECTRON

747 FLIGHT SIMULATOR

Superbly realistic instrumentation and pilot's view in lifelike simulation which includes emergencies such as engine fires and systems failures. This program uses high resolution graphics to the full to produce the most realistic flight-deck display yet seen on a home computer. There are 21 real dials and 25 other indicators (see diagram). Your controls operate throttle, ailerons, elevators, flaps, slats, spoilers, landing gear, reverse thrust, brakes, etc. You see the runway in true perspective. Uses joysticks and includes options to start with take-off or random landing approach. "A real simulation, not just another game" *(Your Computer, April 19 '83).*



ACTUAL SCREEN PHOTOGRAPH

Cassette £9.95 (p&p and VAT included):

DACC Ltd (Dept. DU)

23 Waverley Road, Hindley,
Greater Manchester WN2 3BN.
(Despatch within 48 hours)

In US $27.95 (p&p included)
Order from sole distributor:
Frank Ashton (Dept. DU)
PO Box 7037
Chula Vista CA 92012 7037
(California residents add 6% sales tax)

M & J Software

DRAGON ADVANCED PROGRAMMER'S PACKAGE £15
A cassette-based implementation of FORTH which now includes an important addition enabling the language to be switched between 32K and 64K versions. Complete with a macro-assembler and powerful editor we believe this package to be the most comprehensive on the market. Not only do you have the advantages of FORTH's speed but you can also access Basic's commands from within FORTH programs. Also included completely free is a copy of the fig-FORTH Installation Manual.
To make the package complete we also include a copy of the DRAGON COMPANION.

DIY FORTH for the DRAGON £12
A 6809 fig-FORTH assembly listing and Installation Manual — all you need to put FORTH on to your Dragon. The language is many times faster than Dragon basic, so well worth the effort involved.

ALL ABOUT FORTH by Haydon £7.95
An excellent reference book with cross references to fig-FORTH, the FORTH-79 standard and Starting FORTH. This book should be next to every FORTH programmer's computer. (270+ pages).

Cheques and POs to: M & J SOFTWARE, 34 GRAYS CLOSE, SCHOLAR GREEN, STOKE-ON-TRENT, ST7 3LU. Tel: (0782) 517876.

THE TYPIST

TURNS YOUR DRAGON INTO A SIMPLE WORD PROCESSOR!!!

You can now write professional quality letters on your printer. Your keyboard becomes a typewriter with full upper and lower case characters. Features include full on screen editing, change, insert or delete any character will file or retrieve data from mass storage medium suppresses word wrapround automatically, plus much more. Tape version **£6.95.** Delta disc version **£9.95**

Plus Pontoon. Can you beat the dragon? Full colour graphics, both hands on screen. Recognises pontoons, five card tricks, with automatic ace evaluation.

And Hi-Low based on the popular TV game, can you predict the next card. Very hard, very addictive

Pontoon and Hi-Low are on tape and cost **£4.95** each
Our prices include post and packing and VAT

SOUTH MOLTON COMPUTERS

DOOTSON HOUSE, SOUTH MOLTON, NORTH DEVON, EX36 4EJ

```
0 ELSE 1480
1410 IF Z$="H"THEN IF V=H THEN 142
0 ELSE 1480
1420 A$="RIGHT":GOSUB3000:C(I,J)=1
:K=K+1:IF C=1 THEN SC=SC+1
1430 FOR M=1TO3:LINE((I-1)*21+1,(J
-1)*10+1)-((I-1)*21+20,(J-1)*10+9)
,PRESET,BF
1440 PLAY"V31T60EFGABEFGABEFGAB"
1450 I$=STR$(H):IF LEN(I$)=3 THEN
X=6 ELSE IF LEN(I$)=2 THEN X=9 ELS
E X=3
1460 DRAW"BM"+STR$((I-1)*21+X)+","
+STR$((J-1)*10+8):GOSUB4000
1470 NEXT M:GOSUB2000:GOTO1100
1480 IF VAL(B$)=0 AND C<3 THEN1500
 ELSE IF VAL(B$)=0 AND C=3 THEN GO
SUB2000:GOTO1510
1490 SOUND1,5:A$="WRONG":GOSUB3000
:GOSUB2000:IF C=3THEN1510
1500 DRAW"BM88,145":A$="TRY@AGAIN"
:GOSUB3000:GOSUB2000
1510 GOSUB1600:GOTO1320
1600 REM**clear*wrong*ans**
1610 IF Z$="I"THEN LINE(88,160)-(1
00,150),PRESET,BF
1620 IF Z$="J"THEN LINE(112,150)-(
124,160),PRESET,BF
1630 IF Z$="H"THEN LINE(135,160)-(
160,150),PRESET,BF
1640 IF C=3 AND Z$="I"THEN GOSUB20
00:I$=STR$(I):DRAW"BM88,160":IF LE
N(I$)<3 THEN X=94:DRAW"BM"+STR$(X)
+",160"
1650 IF C=3 AND Z$="J"THEN GOSUB20
00:I$=STR$(J):DRAW"BM112,160"
1660 IF C=3 AND Z$="H"THEN GOSUB20
00:I$=STR$(H):DRAW"BM135,160"
1670 IF C=3 THEN GOSUB4000:DRAW"BM
90,145":A$="RIGHT@ANS":GOSUB3000:P
LAY"T60O5V31EFGAB":GOSUB2000:GOTO1
150
1680 RETURN
2000 REM**hold*and*clear*box**
2010 FOR D=1TO1000:NEXT:LINE(88,14
5)-(150,140),PRESET,BF
2020 IF K=121THEN 2040 ELSE RETURN
2030 REM**draw*score**
2040 PCLS:DRAW"BM100,40":A$="BOARD
@FULL":GOSUB3000
2050 DRAW"BM60,60":A$="YOU@SCORED@
":GOSUB3000:I$=STR$(SC):GOSUB4000:
A$="@OUT@OF@":GOSUB3000:I$=STR$(K)
:GOSUB4000
2060 DRAW"BM70,80":A$="ON@YOUR@FIR
ST@TRY":GOSUB3000
2070 MIN=FIX((TIMER/50)/60):SEC=FI
X((TIMER/50)-(MIN*60))
2080 DRAW"BM67,100":A$="AND@TOOK@"
:GOSUB3000:I$=STR$(MIN):GOSUB4000:
A$="MIN@":GOSUB3000:I$=STR$(SEC):G
OSUB4000:A$="SEC":GOSUB3000
2090 DRAW"BM60,120":A$="ANOTHER@GO
@YES@OR@NO":GOSUB3000
2100 A$=INKEY$:IF A$="Y" THEN RUN
ELSE IF A$="N"THEN POKE65494,0:END
 ELSE2100
2500 REM**draw*digit*number**
2510 B$=" ":TI=0
2520 C$=INKEY$:IF C$=CHR$(13) THEN
 V=VAL(B$):RETURN
2530 TI=TI+1:IF TI=80 THEN V=VAL(B
$):IF V=0 THEN DRAW"BM90,145":A$="
TOO@LATE":GOSUB3000:SOUND5,5:GOSUB
2000:RETURN:ELSE RETURN
2540 IF C$="E" THEN PCLS:GOTO2050
2550 IF Z$="I" AND B$=" " THEN DRA
W"BM91,160"+N1$(11):FOR D=1TO10:NE
XT:LINE(100,150)-(88,160),PRESET,B
F
2560 IF Z$="J" AND B$=" " THEN DRA
W"BM112,160"+N1$(11):FOR D=1TO10:N
EXT:LINE(124,150)-(112,160),PRESET
,BF
2570 IF Z$="H" AND B$=" " THEN DRA
W"BM140,160"+N1$(11):FOR D=1TO10:N
EXT:LINE(160,150)-(135,160),PRESET
,BF
2580 IF Z$="I" OR Z$="J"THEN IF C$
<"0" OR C$>"9" OR LEN(B$)=3 THEN 2
520:REM**two*digit*number**
2590 IF C$<"0" OR C$>"9" OR LEN(B$
)=4 THEN 2520:REM*three*digit*
2600 B$=B$+C$:DRAW N1$(VAL(C$))+"B
R3":GOTO2520
3000 REM**draw*words**
3010 FOR A=1TOLEN(A$):DRAW AZ$(ASC
(MID$(A$,A,1))-63)+"BR3":NEXT A:RE
TURN
4000 REM**draw*numbers**
4010 FOR A=2TOLEN(I$):N=VAL(MID$(I
$,A,1)):DRAW N1$(N)+"BR3":NEXT A:R
ETURN
5000 REM**instructions**
5010 CLS:PRINT @74,"instructions"
5020 PRINT @128,"LEARN YOUR 2 TO 1
2 TIMES TABLES"
5030 PRINT @161,"SECTION:-1......A
*B=?"
5040 PRINT @200,":-2......?*B=C"
5050 PRINT @232,":-3......A*?=C"
5060 PRINT @264,":-4......RND(1,2,
3)
5070 PRINT @289,"INPUT WHICH SECTI
ON YOU WISH TO FIND '1' OR '2' OR
'3' OR '4'":PRINT @365," ";:INPUTZ
5080 IF Z<1 OR Z>4 THEN 5000
5090 PRINT @365,"**"Z"**"
5100 IF Z=4 THEN ZZ=4
5110 GOTO5150
5120 IF Z$="1"THEN Z$="H"
5130 IF Z$="2"THEN Z$="I"
5140 IF Z$="3"THEN Z$="J"
5150 PRINT @420,"press enter to co
```

continued on page 56

```
ntinue"
5160 ZZ$=INKEY$:IF ZZ$<>CHR$(13)TH
EN 5160
5170 CLS:PRINT @74,"instructions"
5180 PRINT @132,"INPUT THE MISSING
 NUMBER                  AND PRESS ENTER
"
5190 PRINT @194,"YOU HAVE THREE TR
YS AND A TEN     SECOND TIME LIMIT
EACH TRY"
5200 PRINT @292,"press enter to co
ntinue"
5210 ZZ$=INKEY$:IF ZZ$<>CHR$(13)TH
EN 5210
5220 CLS:PRINT @234,"PLEASE WAIT"
5230 RETURN
```

Tandy to Dragon Converter

From John Buckley in Salford

MY INTEREST in writing this program began when I bought the July 1983 edition of the American magazine *Rainbow*. This issue contained a free record of three programs but these had been recorded in Tandy format and although I had no problems loading these into my Dragon they would not run.

On listing the programs the reason for this became obvious, as most of the lines appeared to be nonsense. I was puzzled by this but an article in the same issue of *Rainbow* made things a little clearer. I discovered that on both the Tandy and the Dragon, basic key words such as PRINT, FOR, RESET are stored in memory as a one byte token.

This saves memory and also makes a Basic program execute faster. The article also contained a table of the Tandy key words and their corresponding token. I thought therefore that the Dragon tokens could be different.

I then found an article in the August issue of *Dragon User* by Rodney Jones and by using the method he described I was able to to obtain a list of the Dragon token and I found as I had suspected that many of them were different from the Tandy.

The article in *Dragon User* also described in detail how the Dragon organises and stores its program and so I now had all the information I needed to attempt to write a conversion routine. Incidentally I found that a function such as INT, SQR, PEEK is stored as two bytes but the first byte is always $FF.

What I required therefore was a routine which would look at each byte of the program in turn, check if it was a token and if so replace it with the corresponding Dragon token. Being a newcomer to machine code I am sure that my approach is not the most elegant possible but it does seem to work and takes no time at all even for a very long program.

I wrote the routine with the aid of the Dream Editor/Assembler from Dragon Data.

The Basic loader program is given in Listing 1. Type this in and save it on tape before you run it as the program destroys itself in the last line. The program checks for errors in the data statements and will stop if it finds any.

If all is well the message "TANDY-DRAGON CONVERTER READY" will be displayed when the program is run. You can now load a Tandy tape and then type as a direct command EXEC 32580 (followed by ENTER of course) and you should find that the program has been converted to Dragon format.

Please note that only the key words are changed and that the value of PEEKS or POKES remain unchanged so you will have to look through the program and alter these by hand (using EDIT) if necessary.

Program notes

The routine works by first finding the start and end addresses of the program to be converted (location 25 stores the high byte of the start address and location 26 stores the low byte. The end address is two bytes less than the contents of locations 27 and 28).

It then looks at each byte of the program in turn, skipping over line numbers and next line pointers, and checks if it is a token (value $80 or higher) and if so it finds the corresponding Dragon token from a look-up table and places this value at that point in memory.

If the value of a byte is $FF then this indicates a function and there is a second look-up table to deal with these. Once each byte of the program has been dealt with in this way control is returned to BASIC.

The table shown is the "Tandy to Dragon Conversion of keyboard press table, peeks 338-345."

TANDY	DRAGON
254	251
253	247
251	239
247	223
239	254
223	253
191	191

```
10 CLEAR200,32579
20 FORX=32580TO32766
30 READA$:Z=VAL("&H"+A$):CKSUM=CKSUM+Z:POKEX,Z
40 NEXTX
50 DATA34,36,9E,1B,30,1E,BF,7F,68,9E,19,30,04,BC,7F,68,24,14,A6,84
60 DATA81,80,24,11,81,00,27,04,30,01,20,F2,30,01,20,E7,00,00,35,36
70 DATA39,10,8E,7F,8F,E6,84,C1,FF,27,08,C0,80,A6,A5,A7,84,20,E1,10
80 DATA8E,7F,DD,30,01,E6,84,C0,80,A6,A5,A7,84,20,D1
90 DATA80,81,82,83,84,85,86,87,88,89,8A,8B,8C,8D,8F,90,91,92,93,94
100 DATA95,96,97,99,9A,9B,9C,9D,9E,9F,A0,A1,A2,A3,A4,A5,BB,BC,BD,BF
110 DATAC0,C1,C2,C3,C4,C5,C6,C7,C8,C9,CA,CB,CC,A6,A7,A8,A9,98,8E,AA
120 DATAAB,AC,AD,AE,AF,B0,B1,B2,B3,B4,B5,B6,B7,B8,B9,BA,BE,CD
130 DATA80,81,82,A1,84,88,8C,8D,8E,8F,90,91,92,93,96,97,98,99,9A,9B
140 DATA8B,89,8A,87,94,86,83,85,95,9C,9D,9E,9F,A0
150 IF CKSUM<>25580 THEN PRINT"DATA ERROR":END
160 CLS:PRINT"TANDY-DRAGON CONVERTER READY"
170 NEW
```

SOFT SHOP

ATTENTION DRAGON OWNERS!

Increase your Dragon power to 48K with our Ram Packs.

		Please Tick
8K Ram Pack	£34.95	☐
16K Ram Pack	£44.95	☐
One Ribbon Cable	£5.95	☐
Motherboard 1 (8 edge connectors)	£54.95	☐
Motherboard 2 (8 edge connectors)	£54.95	☐

and please send me more information on increasing my Dragon to 256K and 32K (please enclose SAE) .... ☐

All prices include VAT.

Name ....

Address ....

....

I enclose cheque/PO for £........... made payable to SOFT SHOP.
Send orders to (enclose £1.00 p&p):
SOFT SHOP
18 Hothan Close, Hurst Park, Molesey, Surrey. (Tel: 01-979 1078)
Please allow 28 days delivery. All items 12 months guarantee.

FOR THE DRAGON 32
ACE HIGH (MK 2)
Machine gun warfare in the sky

Your machine — The Spitfire, eight machine guns, 16 one-second bursts.
Your target — slow moving Heinkel bombers escorted by formidable Messerschmitt fighters.
Your task — take the bombers apart. Engage the Messerschmitts only if they get in your way.
A high score gives you a second run and a crack at ACE qualification — 25 raiders.
The alert pilot survives. Accurate aim scores high.
Up to ten minutes violent action.
Features hi-res graphics, cockpit view, rear mirror, 4 levels, realistic sounds, reward system and detailed scorecard.
A quality game using all 32K.
Not available in or to libraries or club systems.
Fast service **£6.95**
TUDOR WILLIAMS
15 Summerhill Road, Bilston, W. Midlands WV14 8RD

OMEGA SYSTEMS

DRAGON 32 PROGRAMS

PONTOON — For 1-4 Players **£3.95** inc.
Hi-Res card game, using full 32K. Twist, Stick or Buy from our Dragon Banker.

MENU — Why buy expensive discdrives?
Try our menu-driven tape and find any of our **8 FREE** example programs in seconds!!!! Including Minichess, Blockbusters, Galactica, Grand Prix, etc. Use our menu to save or load up to 33 of your own programs. **£3.95** inc.

OFFER — Send your own clear tapes and pay only **£2.50** inc each or **£4.90** both programs (minimum size C15 tapes).
All overseas enquiries please add £1.00 p&p.
CHEQUES/POs PLEASE TO:
OMEGA SYSTEMS
44 Curlew Road, Abbeydale, Gloucester GL4 7TF



GET TO THE HEART OF YOUR
DRAGON 32



THE MACHINE CODE DEVELOPMENT AID

S22

★ simple to use cartridge
★ includes many features
★ manuals easy to understand
★ co-resident with Basic

An ideal tool for learning MACHINE CODE PROGRAMMING
A permanent addition to your computer's power

ASHBY COMPUTER CENTRE
(UK sole distributors)
186 Ashby High Street, Scunthorpe
South Humberside. Tel: Scunthorpe 871756
Mail order from stock *Trade enquiries welcome*

? ? ? ? ? ? ? ? ? ? ? ? ?

WITCHWAY

A three-part adventure-type maze game for the Dragon

CAN YOU

? open the 12 gates ?
? find the mine entrances ?
? get the clues you need ?
? avoid the mine supervisor ?
? find the witch's key ?
? fight off her police squad ?
? cross the electrified grid ?
? unlock the safe ?
? control the wand ?
? discover the rings ?

"A remarkable effort — more please" *(Dragon's Teeth)*
"Should keep you entertained for many hours" *(Dragon User)*
At **£6.40**, **WITCHWAY** is superb value! Only from **Wellbrooksoft, Dept. DU, The Slade, Peterchurch, Herefordshire HR2 0SU**

? ? ? ? ? ? ? ? ? ? ? ? ?

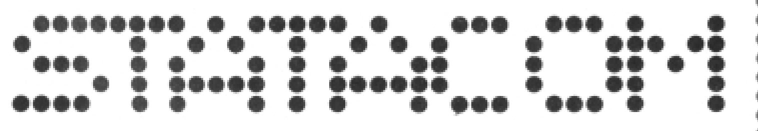
RED ROM DATA

proudly presents two new racing forecasters for the flat season.

SUPERPUNTA
● A specialist program which utilises much of the data supplied in Sporting Life's Weekender paper.
● This most sophisticated program leads you step by step through each runner's form then produces the probable race placings.
● Includes facility to alter significance of any/all factors if required.
£7.95 inclusive

COMPUNTA
● A more general version of the forecaster program for use with any daily paper's racing selection.
£5.50 inclusive

Easy to use — No records to update
Both programs available for:
SPECTRUM 48K, DRAGON 32/64, TANDY COCO 32K
Available ONLY from:
RED ROM DATA, 72 LAMBERT ROAD, GRIMSBY, LINCS DN32 0NR
STATE MACHINE FOR PROMPT DELIVERY





3" DRIVES
FOR DRAGON 32/64
FOR ONLY £330 incl VAT + P&P

Price includes EVERYTHING needed to operate the drive, including DELTA CONTROLLER or DRAGON DOSS
The drives use the latest technology eg a "flippable" diskette that is virtually indestructible
DEALER ENQUIRIES WELCOME
Please contact:
STATACOM DISTRIBUTION LTD
234 High Street, Sutton, Surrey SM1 1NX.
Tel: 01-661 2266

S.P. ELECTRONICS

DRAGON 64	£225.00
DRAGON 32	£149.00
DISK DRIVE (Single and Dual) controller from ...	£275.00
JOYSTICKS (each)	£7.50
MCP40 4-Colour Plotter Printer	£149.00
DELTA 10X Printer	£275.00
CP80 Dot Matrix Printer	£230.00
PRINTER CABLE (Centronics)	£12.90

OS9 AND ASSOCIATED SOFTWARE
Large selection of new Dragon Data, Salamander, Microdeal software
SAE for free list
All prices include VAT, carriage extra
S.P. ELECTRONICS, 48 Limby Road, Hucknall, Notts
(Nottingham 640377)
Official Dragon Servicer



BARCLAYCARD VISA

Dragon Answers

If you've got a technical question or problem write to Brian Cadge at *Dragon User*. Please do not send a SAE as Brian cannot guarantee to answer individual inquiries.

Disk drives

I AM A Senior Citizen and have purchased a Dragon 32 to help me use my time sensibly. I also have a Datex 3 tape recorder which, after adjustment, gives excellent results.

My main interest is data processing and I am considering buying a disk drive, but I have been put off by articles I have read in your magazine. For example can I expect true random search and is there a danger of losing entries?

H McDonald
Bexhill-on-Sea
East Sussex

ALL DISK drives are, by their very nature, capable of true random access, what matters to the Basic programmer is how well the Basic supports this. The two disk systems available for the Dragon (Premier and Dragon Data) both support serial and random access files albeit with different commands and syntax, so it is really a matter of personal choice which system to go for.

There is no danger of losing entries by accident (except physical damage to the disk itself) as disk drives are much more reliable than cassettes. Also both systems mentioned above have the facility to verify all data, as it is saved to disk as a precaution and to produce back-up files.



Crossed wires

I FIND it hard to understand the wiring of my joystick. I've bought an ordinary Atari joystick and dismantled the plug. When I dismantled it I didn't know which wire was for firing, or what the other wires were for.

I would be grateful if you would put an illustrated diagram with instructions in the next issue of *Dragon User*.

Cheuk Leu
Bedminister Down
Bristol

I'M AFRAID I can't help you too much with this one. The Dragon's joystick ports are designed for use with potentiometer-type joysticks as opposed to switch type. The Basic reads the joystick position by the voltage returning from the port compared to that sent out. Atari-type joysticks are simple on/off switches and cannot be connected directly to the Dragon. Several interfaces are available — for example, from Mr Micro, 69 Partlington Lane, Swinton, Manchester M27 3AL and Cotswold Computers, 6 Middle Row, Chipping Norton, Oxfordshire.

Unless one understands exactly how the two types of joystick operate there is no simple rewiring job that can be done yourself.

Planting seeds

MY FIRST problem was how to access the various graphics modes using machihe code, so I was happy to find the answer in *Dragon User*.

However, my problem now is how to generate random numbers in machine code. Is there a subroutine in the Basic Rom and if so what is the address?

Until I hear from you the aliens will keep coming from the same location.

Dave Smith
Sutton Park
Hull

THERE ARE many ways of generating random numbers in machine code, they all depend on "seeds" and the same set of "random" numbers will be produced if the starting value of the seed is the same each time.

The following assembly language routine will return a "random" number in the "B" register and assumes that SEED is a two-byte location which was set up with a suitable number (eg value of timer from location 274) at the start of the main program. This method is not very sophisticated, but is short and suitable for space invader type programs.

```
500 RAND   PSHS A
510        LD D SEED
520        ADD D #3
530        ST D SEED
540        MUL
550        MUL
560        MUL
570        PULS A,PC
```

Altering amplitude

I HAVE come across a few articles on how to set up the registers for sound in machine code, but nothing on addressing the amplitude section in machine code, which would enable more interesting amplitude changes to be achieved.

I find using:

```
FOR V = 31 TO 1 STEP 2
x$ = "V" + STR$(V)
NEXT
```

too slow. How can I address change in sound level?

Richard Brooks
Croydon
Surrey

TO ACCESS sound in machine code, bit 3 of locations $FF01 and $FF03 must be cleared and bit 3 of location $FF23 must be set. The D/A convertor which is now set up for sound is addressed as the 6 MSB of location $FF20. These six bits control the amplitude of the signal sent to the speaker. Therefore the higher the number stored here the higher the amplitude of the sound. Being a six bit D/A convertor there are effectively 64 different volume settings.

When using location $FF20 make sure that bit 1 remains at zero as this is the printer strobe connection and will cause any printer attached to behave unpredictably if left high. For the sake of completion, bit 0 of this location is used as the cassette data input.

Bits and pieces

COULD YOU please advise me on the following problems.

Why is it that although the Dragon has 24871 bits of memory available to the user I cannot clear 16K for machine code, eg CLEAR 16000, any address without an OM error?

How can I transfer a machine code program from an address higher than the address I wish it to start, ie you cannot CLOADM with a negative offset.

I wish to disable the reset button and the break key, and leave the rest of the keyboard operative. Can you tell me the relevant POKEs?

D Pendleton
Castlefields
Runcorn

YOUR FIRST point is quite simple. The first number in a clear statement is the amount of string space to be cleared and then the second number sets the highest memory location to be used by Basic. So to clear 16K of memory you would type CLEAR 200,16384. In fact, clear 16000 will work and clear 16000 bytes of string space provided that the Basic program isn't too large.

Negative offsets can be used in a CLOADM but not directly. For example, if the offset was to be −3072 bytes, you need to type CLOADM "", 65536-3072. The 66536 is the important part — this produces a positive offset which will have the same effect.

The third point, disabling break and reset, is not so simple. In a previous issue it was explained how to disable break, but the reset button cannot be ignored as this is a physical connection to the CPU. It can, however, be redirected — locations 114/115 point to the address to jump to when reset is pressed. The first instruction must be a NOP, this could be followed by JMP 33951 which would run a program if reset is pressed.

Little Brothers should be seen but not heard.

brother THERMAL TRANSFER PRINTER HR-5

REGULAR, CONDENSED, OR EXTENDED FACES.

CUT SHEET A4 OR ROLLER PAPER.

BATTERY OR MAINS OPERATED.

A maxim which eloquently describes the Brother HR-5.

Less than a foot across, it's nonetheless loaded with features.

But there's one thing the HR-5 won't give you. Earache.

For the annoying 'clickety clack' many printers produce is mercifully absent from the HR-5.

Quietly efficient, it delivers high definition dot matrix text over 80 columns at 30 c.p.s.

The HR-5 also has something of an artistic bent.

Being capable of producing uni-directional graph and chart images together with bi-directional text.

It will also hone down characters into a condensed face, or extend them for added emphasis.

Incorporating either a Centronics parallel or RS-232C interface, the HR-5 is compatible with most home computers and popular software.

Perfectly portable, the battery or mains operated HR-5 weighs less than 4lbs, and has a starting price of only £179.95 (inc.VAT).

Which is really something to shout about.

PLEASE SEND ME MORE DETAILS OF THE REMARKABLE BROTHER HR-5 PRINTER.

DU6/84

NAME_______________

ADDRESS_______________

____

____

_______________ TEL.NO._______________



DEPT P, BROTHER OFFICE EQUIPMENT DIVISION, JONES + BROTHER, SHEPLEY STREET, GUIDE BRIDGE, AUDENSHAW, MANCHESTER M34 5JD
TEL: 061 330 6531 (10 LINES) 061 330 0111 (6 LINES) 061 330 3036 (4 LINES). TELEX: 669092. BROTHER INDUSTRIES LTD., NAGOYA, JAPAN.

Classified

DUST COVERS

★ DRAGON 32 & 64 ★

made-to-measure in quality natural vinyl with cut-outs for terminals and leads. Just send **£2.95** (no stamp required) to:

ALLEN ENTERPRISES
Freepost, Luton LU2 8BR
Trade enquiries welcome

DRAGON MICRO
TOP QUALITY JOYSTICKS

At an amazingly low price. Only **£9.95** a pair, including P&P. Easier to handle and faster than others costing twice as much.

Cheques/POs to:
PERITRON (DEPT DU)
21 Woodhouse Road, London N12 9EN

DRAGON HITEXT. Place text on the hi-res screen with this machine language utility. Features full upper, lower case, reverse field and double height characters. 224 definable, 24 x 32 display, maximum 10 cls colours, six display modes, £4.95 to: R. Thompson, 7 Brindley Way, Southall, Middlesex UB1 3JN. Tel: 01-571 3610 (evenings).

NEW PEN PAL CLUB for Dragoneers. Exchange, advice, notes, ideas, swap software, hardware with friends. Interested? Details SAE. Belfield, Stour Hill, Quarry Bank, Brierley Hill, West Midlands.

MACHINE CODE PROGRAMMERS DATA PACK contains the data booklets for the Dragon's 4 main ICs (6809, 6821, 6847 and 6883) plus a 6809 Reference Card. Over 80 pages of software and hardware data. **£3.95**

'THE 6809 COMPANION' is a valuable reference book for the 6809 programmer. Excellent value at only **£2.10**

SPECIAL OFFER. Buy both for only **£5.50** Prices include p&p. Send a stamp for details of over 70 items for your Dragon.

ELTEK, 29 PARKSIDE AVENUE, LITTLEHAMPTON, WEST SUSSEX, BN17 6BG.

ANYONE FOR CRICKET?
1 joystick required ★ Save game ★ 4 match series ★ 10 bowler options

WORDSEARCH
Great family game ★ 300+ words in 10 categories ★ 1 to 4 players ★ different game each round

FILE-IT
16,000 characters in 500 files ★ menu driven, printer output — £4.95 each, any 2 for £8, all 3 for £11 p&p inc.
Basic listings 50p per program. Send sae prompt return on all orders.

DORSOFT, 15 Gurjun Close
Upton Poole, Dorset BH16 5QH

DRAGON 32 SOFTWARE

1. MATHS PACK ONE **£2.99**
2. ENGLISH TEST **£2.99**
3. MEASURES TEST **£1.80**
4. AMERICAN CAPITALS* **£1.80**
5. MULTI-MATHS **£1.80**
6. POKER* **£2.99**
7. HANGMAN* **£2.50**
8. CHEMISTRY TEST **£1.80**

*Features full colour High Res Graphics. Send cheques/POs to: P. Perris, 88 Poplar Crescent, Shipley, West Yorkshire BD18 2HH, or sae for full details.

COVERS. Dragon 32 high quality vinyl covers with lead terminal cut-outs. Send £2.95 to Warwick Trim, Unit 4, Hurlbutt Road, Heathcote, Leamington Spa.

BASIC AUTO-RUN PROGRAM, £4.50. Generation X Software, 9 Park Road, Batley, West Yorkshire, WF17 5LP. Delivery by first-class return post.

GAME PACKS
Game Pack One featuring Bomber, Hangman with 400 word vocabulary, Simon.
Game Pack Two featuring Car-race, Breakout and Snake.
All above entirely in Hi-res, EVEN TEXT!!! Prices **£5.95** each or both for **£10.00.**
Send cheques/POs to:
Ian Elkington, 11 Wharfedale Gardens, Baildon, Shipley, West Yorks BD17 6TN

AND NOW FOR THE DRAGON 32 . . . Count Krakup's Castle. A top quality two-part adventure over 50K long! In part one. Will you escape from the Count's dungeons? With over 500 rooms in full colour, 3D high resolution graphics and sound. In part two. Return the amulet and join the Count's family. Can you succeed against all odds? To find out send a cheque or postal order for £5.95 made payable to: T. Cartledge, 39 High Street, Chellaston, Derby DE7 1TB.

FOUR DRAGON PROGRAMS, two arcade style games, two utilities including machine routines, £3. 203 Hawthorn Road, Strood, Kent.

CASSETTE FORMAT PROGRAM. Fast access any program. Catalogues loads, saves, reviews tape contents, menu driven. £2 for listing. SAE details: R. G. Staplehurst, 13 Overdale, Bletchingley, Surrey RH1 4AG.

MAKE YOUR BASIC PROGRAMS AUTO-RUN upon loading while displaying an easily designed screen. Full instructions included. Basic Auto-Run on cassette. From Generation X Software, 9 Park Road, Batley, West Yorkshire, WF17 5LP. (Cheques/POs payable to J. Halliley.) Delivered by first-class return post. Unbeatable value at only £4.50.

DRAGON CLONE-IT will back-up, copy all machine code programs up to 30K, including Cuthberts 100% machine code, £6. K. Hunter, 7 Balladen, Rossendale, Lancashire.

DRAGON SOFTWARE LIBRARY. Life membership only £6. Two weeks' hire £1.25. SAE for details to: DSL, 30 Waterside View, Leysdown, Sheerness ME12 4PW.

ROUTINES MAKING PROGRAMS. Programs unlistable, unbreakable, unsavable, etc! Cheque/PO £1 to: A. J. Wass, 19 Fernwood, Marple SK6 5BE.

COMPUTER CORRESPONDENCE COURSES (INTERNATIONAL)
A complete course in BASIC programming written specifically for the DRAGON computer.
For details write to: **Computer Corr. Courses (Int.), Churchway, West Ilsley, Nr. Newbury, Berks. Tel. (063528) 336**

MICRODEAL TELEWRITER WORD PROCESSOR, software with instructions, £38, also original tapes, cartridges available. Tel: Leeds 860887.

DRAGON LISTINGS from cassette by return post, first copy £1, second copy 50p. K. Copeland, 23 The Lane, Awsworth, Nottingham. Tel: 0602 324658.

FREE SOFTWARE CATALOGUE available from LB Games containing one of the largest ranges of software for the home computer. No SAE required. Send to LB Games, PO Box 410, Ascot, Berks SL5 0JA.

PRINT THAT PICTURE. Dragon to Acorn/Seikosha GP100A, Tandy 100/120/200, fast machine code, £5. Disassembler to screen or printer, ASCII and hex, in Basic, £5. As above in relocatable machine code, £5.95. ASP, 6 Monks Leys Trerrace, Lincoln LN2 5JG.

CENTRONICS LEADS for the Dragon, 1.5m long to fit all standard Centronics interface printers, £13.50 inclusive VAT, postage. Quick despatch. cheques payable to: High Tech Developments, 27 Quantock Road, Weston-super-Mare, Avon BS23 4DD.

DISCOUNT DRAGON SOFTWARE
Top quality programs at low prices
e.g. **Dragonchess** (Oasis) .... only **£8.95**
Grand Prix (Salamander) .... only **£6.95**
Leggit (Imagine) .... only **£4.75**
Microdeal Programs .... only **£6.95**
prices include p&p (Europe +50p, others +£1.00) (sae for list) **Cheques/POs to: D J SOFTWARE, 273 Mossy Lea Road, Wrighington, Wigan, Lancs WN6 9RN.**

Here's my classified ad.

(Please write your copy in capital letters on the lines below.)

£0.20	£0.40	£0.60
£0.80	£1.00	£1.20
£1.40	£1.60	£1.80
£2.00	£2.20	£2.40
£2.60	£2.80	£3.00
£3.20	£3.40	£3.60
£3.80	£4.00	£4.20
£4.40	£4.60	£4.80
£5.00	£5.20	£5.40
£5.60	£5.80	£6.00

Please continue on a separate sheet of paper

I make this.......................... words, at 20p per word so I enclose ....

Name..

Address..

.... **Telephone** ....

Please cut out and send this form to: Classified Department, *Dragon User*, 12-13 Little Newport Street, London WC2R 3LD

Classified

THE DRAGON COMPOSER from Microdeal, £6.50; Textstar from PSS, £6.50; Sprint from Oasis, £6.50. All three for £18.50. Tel: (0823) 83902.

THREE-IN-ONE GAMES CASSETTE. Darts, Connekt 4, Pakman, £3.50. Cheques/PO to: J. Fenton, 10 Ashlar Close, Chell, Stoke-on-Trent ST66UP

"COMPOSER"-PROGRAM DEVELOPMENT UTILITY. This relocatable machine code program gives extra power for typing in basic programs. Automatic line numbering, keyboard auto repeat. Shift key and letter gives basic keywords (or any string of up to eight characters), key expansions easily modified. Only £4.95. P. Canham, 3 The Circuit, Cheadle Hulme, Cheadle, Cheshire.

PROTECT AND AUTO-RUN YOUR BASIC PROGRAMS

A cassette based program which allows you to design your own load screen, then protects and auto-runs your Basic or M/C programs.

MACHINE CODE PROGRAMMING

If you can write basic you can write machine code. Our booklet together with taped routines and DISA our 6809 disassembler will show you how — from the beginning. On the same tape M/C EXEC to auto-exec your programs.

Auto-run tape, £4.00
Machine code package £7.00

WARMSOFT
44 Balfour Road, Walmer, Deal, Kent

DRAGON 32/64 INTERFACE

A new, low-cost, educational, multi-purpose interface system for data acquisition and process control.
Designed and developed to a high professional standard for use at home, in schools, colleges, universities, etc.
Numerous applications around the home, in the classroom and laboratory. Learn about the mysteries of microprocessor control, data acquisition and general interfacing techniques.
Many facilities available, full instructions included in manual, along with Basic and M/code routines for use and numerous application notes.

Available mail order only
Price **£39.95**
(plus **50p** p&p + VAT)
or for more details contact:
ELECTROANALYTICAL TECHNOLOGY
8 Access Road, West Derby, Liverpool L12 4YN

DRAGON SOFTWARE AVAILABLE, SAE to P. L. Buckett, 39 Cromwell Road, Cheltenham, Gloucester for list.

G4BMK RADIO SOFTWARE
DRAGON 32/64
TAPE CARTRIDGE AND DISK

RTTY TRANSCEIVE. Any Baud rate, type ahead etc. Needs no TU for reception.
MORSE TRANSCEIVE. Decodes and sends morse to 150 wpm.
MORSE TUTOR. Any speed, any spacing. Pure tones.
AMTOR SYSTEM. Available soon.
Send large SAE for full details and software/hardware price list.

GROSVENOR SOFTWARE (D)
22 Grosvenor Road, Seaford
E. Sussex BN25 2BS
(0323) 893378

UPGRADE your Dragon 32 to 64K. Includes hardware, screen invert switch Basicopy and utility software. Only £55 with rebate for chips not replaced. Libra Home Entertainment, 307 Edenfield Road, Cutgate Shopping Centre, Rochdale, Lancs.

THREE QUALITY ASTRONOMICAL PROGRAMMES with documentation for the Dragon. "Jovian Satellites" with graphics, real time capabilities/"Precession" convertor/"RA-DEC to ALT/AZ" convertor. On one cassette for £3.85 or £5.95 for Dragon disk. Available from P. A. Jeanes, 112 Priory Avenue, Taunton, Somerset, TA1 1YB. Send PO/cheque.

DRAGON PRINTER LEADS. Fit standard centronics interface printers, £9.95 inclusive. Cheques payable to: R. D. Vowles, 3 Orchard Waye, Uxbridge, Middlesex UB8 2BN.

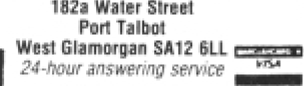
Hi-Res DRAW (artist designer)

Functions include draw, paint, line, box, circle, parts of circle, rubber, clear screen, text merge, help screens, save to tape, load from tape, copy parts of screen, screen dump to MCP-40/CGP-115, user friendly. **£5.95.**
Screen Dump — to MCP-40 or TANDY CGP-115, two sizes plus colour or black and white picture. **£4.50.**
Send cheques/POs to: **Ian Elkington, 11 Wharfedale Gardens, Baildon, Shipley, West Yorks BD17 6TN.**

JOYSTICK INTERFACES

ATARI QUICKSHOT — Dragon, Oric, Spectrum, TI/99, Dragon £9.95 POA for others.
COMING SOON: Trackball for the Dragon less than £20

Trade enquiries welcome
JAY DEE COMMUNICATIONS
182a Water Street
Port Talbot
West Glamorgan SA12 6LL
24-hour answering service



SCREEN COPIER. Have hard copies of your screens, using CGP-115. Send £3 to Unique Software, 5 Milton Drive, Poynton, Cheshire, SK12 1EZ.

DRAGON LISTINGS. Two quality copies, £1.25 (single 75p). Send cassette plus SAE (A4 size), to: Mr J. J. Baines, 21 Orchard Drive, Burton, Scunthorpe, DN15 9EF, S. Humberside. Tel: (0724) 721204.

BACK IT UP. Two machine code programs to copy all Dragon machine code programs, including auto-run programs. Cassette £3.50, listing £1.75 from T. Collins, 51 Magdala Street, Belfast BT7 1PU.

★ UNREPEATABLE OFFER ★

Brand New Professional DM printers with **VERY IMPRESSIVE** specifications at less than **HALF PRICE!**

Look at what you get.
- 100 cps Bidirectional printing.
- 80, 96, 132, 137 column print.
- Quad Density Dot Addressable graphics (19,000 dots per sq. inch)
- Adjustable tractor feed **AND** sheet feed **AND** paper roll
- **And MUCH, MUCH more.**

AND THE PRICE . . .
Only £239.25, which includes RS232 interface, all cables, VAT, and carriage. (Printer only £241.50)

REEL COMPUTER SERVICES
30 Edgmond Close, Winyates West
Redditch, Worcestershire
Tel: Redditch 23447
HURRY! Few only at this price

EXPIN. Enter expressions while running without having to DEF FN. Gets round problem of no EVAL or decent VAL. Cassette sent with full details of program and how to merge, £3.50. Also listings at ½p per line plus 40p p&p, if more than 320 lines just send £2. JP Soft, 14 Dugar Walk, Redland, Bristol, BS6 7DH. Tel: (0272) 48777.

CENTRAL HEATING calculator tape gives radiator and boiler sizes. Ideal for business, college and school use. For Dragon. Torwood Educational Tapes, 250 Leicester Road, Markfield, Leicester.

DRAGON SOFTWARE FOR SALE, 25 tapes, seven cartridges including Forth and Monitor, also three books plus issues 1-12 of Dragon User. All half-price. Tel: Upwey 3707.

DRAGON 32 TAPE COPIER

Now you can make back up copies of all your programs, including auto-run machine code games etc. Display shows program name, program type and data blocks. Easy to use — full instructions.

ONLY £3.95

MICROSTAR SOFTWARE
90 Ruskin Drive, Worcester Park, Surrey KT4 8LJ

LISTINGS FROM CASSETTE, 70p per program (+30p p&p). Fan-Fold Stationery copyright respected. Cheque/PO to: Susan, 14 Claybrick Avenue, Hockley, Essex.

ADVERTISERS INDEX

Competition Corner

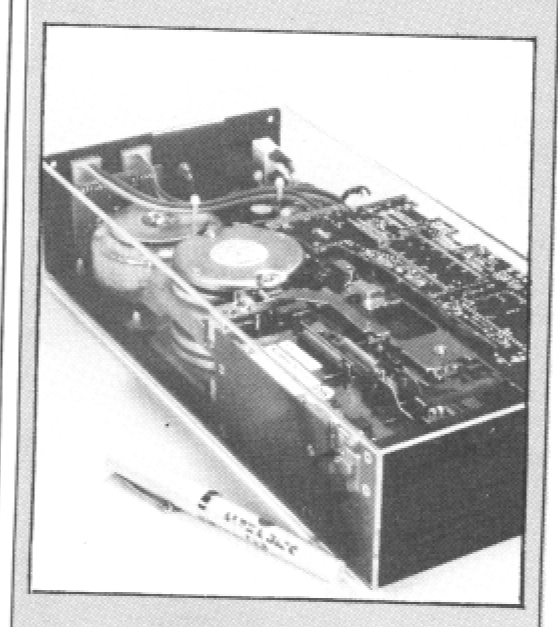
Answers to Competition Corner, *Dragon User*, 12/13 Little Newport Street, London WC2R 3LD

Prize

ALPHA DISC, the UK company featured in our news pages for enhancing Canon's disk drive technology, is giving our readers a chance to win a drive of their own.

The company is offering Canon's MD211 drive to the winner of this month's competition. This is the single density, 40 track version of the 221 featured in our news. The prize comes complete with power supply.

The main difference between the two drives is that the 221 offers a choice between 40 and 80 track disks. But the 200K storage offered on the 40 track 211 should be more then enough for most Dragon users.



Rules

TO WIN the disk drive you must show both the answer to the competition and how to solve it with the use of a Basic program developed on your Dragon. As a tiebreaker complete the following sentence in 15 words or less: "I want a disk drive for my Dragon because . . ."

Your entry must arrive at *Dragon User* by the last working day in June. The winner and solution to the puzzle will be published in our September issue. Entries will not be acknowledged and we cannot enter into correspondence on the result. You may only enter the competition once.

March winner

THE WINNER of March's competition and recipient of a sprite graphics board from Premier Microsystem is B Matthews of Newport in Gwent, who correctly stated that the most favourable choice of number was 176, which resolves to 100 in just four moves.

Win a disk drive

*Alpha Disc provides the prize to **Gordon Lee's** puzzle*

ONE OF my earliest recollections of sport was as a spectator at a mixed doubles tennis match and being a little puzzled by the umpire calling out the score, "Fifteen-love". My first impression was that this was some form of endearment directed towards one of the ladies playing, but as the game progressed and the term seemed to be used somewhat indiscriminately, my confusion grew. It was only when it was pointed out the meaning of the word "love" that things became clearer. Other games too have their own distinctive terms, and if we overheard reference to "one under par", "four faults", "three no trumps", "re-double" and "15-two, 15-four, two for a pair, three for a run and one for his nob" we would know that the games were, respectively, golf, showjumping, bridge, backgammon and cribbage.

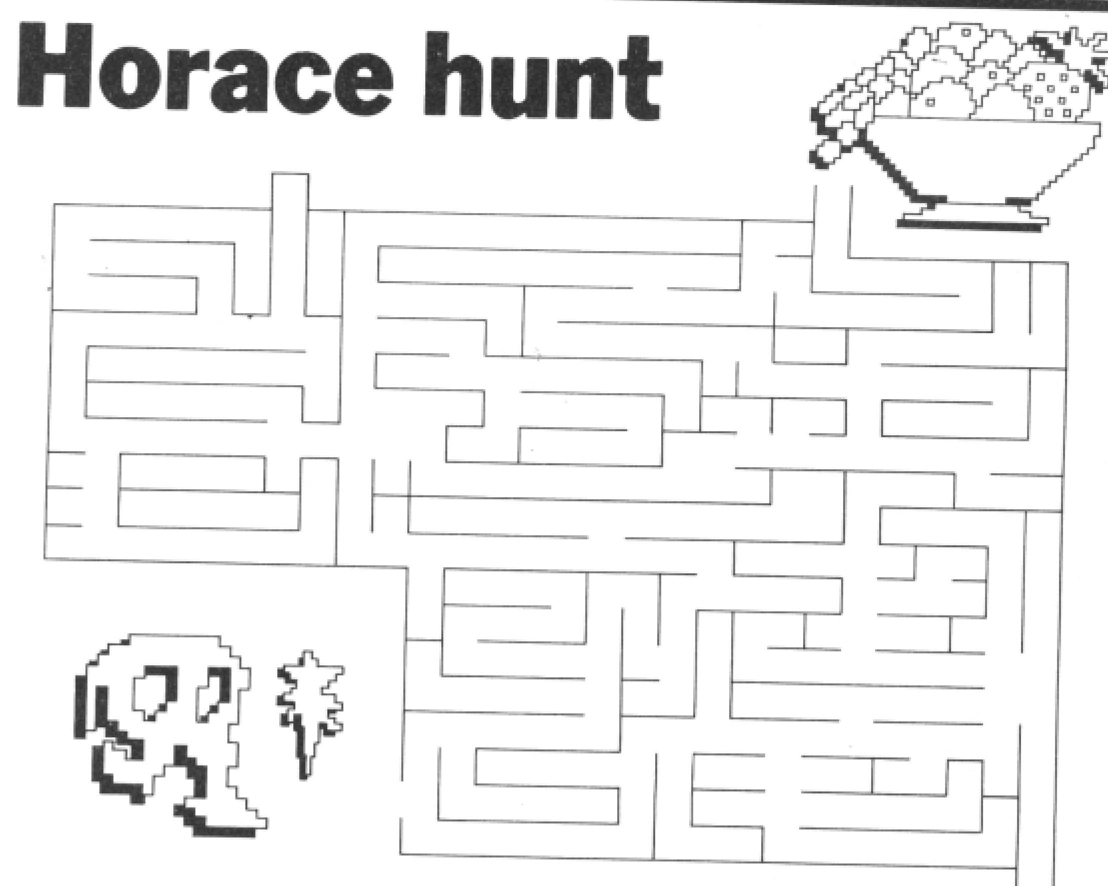
So far we have considered only the means of scoring, but in many games and sports mathematics is more intimately connected. For example, card games involving hands which rank in a set order, such as poker or brag, have that order determined by the statistical expectation of such a hand being dealt. Thus, two pairs are more likely to occur than three of a kind, and so consequently are regarded as the lower hand.

Finally, mention must be made of that most traditional of pub games — darts. It is a game which requires both mental agility in assessing the most advantageous throws especially when nearing the end of the game, and also manual dexterity when it comes to actually throwing the dart. As the game is so closely associated with pubs, it always amazes me that after a couple of pints anyone can play at all!

Consider the following incident which took place in the bar of the Muddlecombe Arms recently. Duggins, who had just thrown three darts, announced that he had landed one in each of three adjacent sectors on the board. Muggins, who was a mathematician, remarked that knowing this, and also knowing the total scored, he could work out exactly where on the board each of the darts had landed.

Buggins, however, overheard this conversation, and even though he didn't know what the total was, he was also able to deduce where Duggins' darts had landed. But then, you see, he did have the advantage of knowing how many doubles had been scored. It was a standard dartboard, and no inner or outer bull was involved. **Where did the darts land?■**

Horace hunt



HORACE is a lovable little rogue who first appeared in a park on the Spectrum, picking the flowers and eating the keepers' lunches. Now Melbourne House is giving 20 of our readers a chance to win a copy of the Dragon version of Hungry Horace.

Just mark in Horace's path through the maze to the food and send the completed entry to us with your name and address attached. As tiebreaker, complete the following sentence in less than 15 words: "I want to own a copy of Hungry Horace because . . ."

Your entries must arrive by the last working day in June and the winners will be announced in our September issue.

St. George now has two choices!

Red or green will slay the Dragon

- **Nylon encased-Steel shafted joystick with ball and socket joint.**
- **Fast sprung return to centre.**
- **Graphite wiper linear potentiometers.**
- **12 Months Guarantee.**
- **7 day Money back Guarantee.**

Complete control at your fingertips

The smooth control of the Voltmace delta sprung return joystick is now available to Dragon owners. Each joystick has been individually tested before it leaves our factory, following extensive robot testing to prove the design (we tested it more than a million times). This means that not only will your joystick be strong, tough and reliable, but it ensures long life, accurate control and with the choice of red or green fire buttons, gives instant recognition between the left and right joysticks.

If you are not completely satisfied with the delta 3d, return it to us within seven days for a full refund.



Dragon dealers come out of your caverns!

More stockists required in some areas.



Made in England

DELTA 3d JOYSTICK £10.00
TWO DELTA 3d JOYSTICKS £19.50
PLEASE STATE COLOURS REQUIRED

Prices include VAT and P&P.

Voltmace delta 3d






VOLTMACE LTD
PARK DRIVE
BALDOCK
HERTS
SG7 6ED
Tel: (0462) 894410

Callers welcome at the factory—Monday to Friday.